FUNGSI ALGORITMA RSA UNTUK MEMODIFIKASI DAN MENINGKATKAN PENGAMANAN ACAKAN BISS

Indra Gunawan[1], Sumarno[2], Heru Satria Tambunan[3]
[1,2,3] *STIKOM Tunas Bangsa Pematangsiantar*
Jl. Jend. Sudirman Blok A, No. 1, 2 dan 3. Kode Pos : 21127
[1]*indra@amiktunasbangsa.ac.id,* [2]*sumarno@amiktunasbangsa.ac.id,* [3]*heru@amiktunasbangsa.ac.id*

Abstrak—**AMIK dan STIKOM Tunas Bangsa yang memiliki keunggulan kompetitif merupakan jantung kinerja lembaga sebuah pendidikan dalam sebuah pasar yang kompetitif. Banyak faktor yang mempengaruhi keunggulan sebuah Akademi Ilmu Komputer. Faktor-faktor tersebut diantaranya merancang dan meningkatkan sebuah keamanan data. Pada saat ini, sudah begitu banyak media-media informasi, promosi, perfilman dan olah raga yang sangat berperan untuk memberikan suatu pengetahuan yang dapat mempengaruhi perubahan di masyarakat. Salah satu contoh media yang sangat berhasil tersebut untuk saat ini adalah media televisi. Dengan adanya media televisi yang dibarengi dengan keluarnya produk-produk digital-reciever yang digunakan untuk sarana penangkapan sinyal siaran melalui satelit, dapat menekankan bahwa media televisi benar-benar dapat dijadikan sebagai patokan untuk sarana penyampaian informasi karena sudah mengacu kepada audio dan visual yang dapat memperjelas suatu keadaan secara langsung. Didalam penyampaian informasinya, media televisi dapat melibatkan media-media yang lain seperti media massa, media cetak, media pendidikan, media olah raga, media perfilman dan media internet sehingga sangat cocok untuk digunakan sebagai media penyampaian informasi kepada masyarakat karena sudah mengacu kepada audio dan visual, yang sekarang ini biasa disebut dengan multimedia. Penelitian ini bertujuan untuk meningkatkan sebuah sistem keamanan suatu jenis acakan dengan memodifikasi algoritma yang digunakan yaitu dengan menggunakan fungsi algoritma RSA untuk memodifikasi acakan BISS agar keamanan data bisa menjadi lebih kuat dibandingkan sebelumnya.**

Keywords—**Keamanan Data, Enkripsi, Dekripsi, RSA, Acakan**

I. PENDAHULUAN

BISS (*Basic Interoperable Scrambling System*) merupakan jenis pengenkripsian yang digunakan untuk mengamankan sebuah video dari suatu sinyal tertentu. *BISS* biasanya berfungsi untuk mengunci beberapa siaran video seperti film-film terbaru yang memiliki hak siar [1]. Dengan menggunakan salah satu jenis acakan diantaranya adalah BISS, maka pengamanan dari siaran video yang premium atau berbayar dapat mengunci atau mengaburkan tampilan dari video tersebut agar tidak dapat diakses oleh orang lain yang tidak memiliki hak.

Seiring dengan kemajuan teknologi, muncullah beberapa *provider* yang menawarkan suatu produk digital-reciever yang mampu menyajikan siaran-siaran premium dan berkualias, baik itu informasi, promosi, perfilman dan olah raga. *Provider* tersebut menawarkan siaran-siaran yang dapat menambah pengetahuan masyarakat. Disamping itu setiap siaran dari sebuah *provider* akan dienkripsi dengan beberapa jenis model acakan, sehingga siaran tersebut sangat mustahil untuk dinikmati oleh masyarakat bila menggunakan digital-reciever biasa atau selain yang ditawarkan oleh provider penyedia. Salah satu jenis acakan yang digunakan adalah acakan *BISS*[2].

Dengan kemajuan media dapat dijadikan sebagai pendorong untuk kemajuan teknologi yang dapat dikombinasikan untuk memberikan suatu informasi kepada masyarakat yang memiliki latar belakang yang beragam, sehingga muncullah beberapa produk dari *digital-reciever* yang mampu untuk membuka enskripsi atau acakan dari jenis acakan *BISS*. Dengan ditemukannya deskripsi dari acakan *BISS* tersebut, banyak pula siaran-siaran premium berbayar tersebut dapat disaksikan dengan cuma-cuma (*free*). Mengacu dari hasil deskripsi acakan *BISS* diatas, penelitian ini memfokuskan kepada pengamanan acakan *BISS* dengan Kriptografi, agar pengenskripsian keamanan acakan *BISS* dapat menjadi optimal.

Algoritma kriptografi RSA dianggap dapat memenuhi tingkat sekuriti yang tinggi. Dengan kombinasi hasil kali 2 (dua) bilangan prima, akan sulit untuk ditemukan dan akan memakan waktu yang sangat lama jika menggunakan *bruteforce*. Kunci RSA dengan panjang 1024 bit akan menghabiskan waktu 1.43 x 10-213 tahun. Waktu yang diperlukan melebihi perkiraan umur alam semesta yang dikalkulasi hanya sekitar 13.75 x 109 tahun [3].

Keamanan acakan BISS pada saat ini sudah sangat mudah untuk dibobol dengan menggunakan *brute force*, bahkan jika sudah mengetahui jenis pola yang

digunakan didalam sebuah acakan BISS, maka tanpa menggunakan *brute force* pun juga bisa untuk membobol jenis acakan BISS dengan menggunakan urutan pola dan kombinasi huruf dan angka.

Berikut beberapa gambar ketika acakan BISS dibobol :

Gbr 1. Siaran Yang Terdeteksi dengan Acakan BISS

Ketika siaran yang terdeteksi dengan acakan BISS didapat, siaran tersebut masih dalam keadaan gelap atau teracak, gambar siaran masih tidak bisa ditampilkan.

Gbr 2. Memasukkan Pola Acakan BISS

Ketika pola acakan BISS sudah diketahui dengan menggunakan 16 karakter digit bilangan hexa, maka karakter tersebut di-*insert* kedalam digital untuk membuka atau menghilangkan siaran yang teracak.

Gbr 3. Acakan BISS Terbobol

Ketika karakter pola dari bilangan hexa sesusai dengan acakan BISS, maka siaran yang teracak pun akan terbuka dengan sendirinya.Oleh karena itu diperlukannya proses pengamanan acakan BISS dengan menggunakan fungsi dari algoritma RSA untuk memodifikasi acakan BISS.

II. TINJAUAN PUSTAKA

Kemanan merupakan masalah besar dan mengamankan data yang penting sangat penting, sehingga data tersebut tidak dapat disadap atau disalah gunakan untuk tujuan ilegal sehingga merugikan pihak lain. Untuk itulah pemerintah dan lembaga lainnya berusaha mengamankan data mereka sekuat tenaga agar tidak terjadi pembobolan. Biarpun begitu tetap saja ada pihak-pihak yang berusaha membobol itu dengan menggunakan berbagai kunci dan juga metode. Untuk menghindari hal tersebut maka data yang dikirim diubah kedalam data yang tidak dapat dibaca oleh sang pembajak dan kemudian data tersebut diubah kembali kedalam bentuk yang bisa dibaca oleh penerimanya. Teknik dan ilmu untuk membuat data yang tidak dapat dibaca sehingga hanya orang yang berwenang yang mampu membaca data, inilah yang disebut dengan kriptografi [2].

Dari sekian banyak algoritma kriptografi dengan kunci-publik yang pernah dibuat, algoritma yang paling populer adalah algoritma rsa. Algoritma rsa yang dibuat oleh Ron Rivest, Adi Shamir dan Leonard Adleman pada tahun 1976. Keamanan algoritma rsa terletak pada sulitnya untuk memfaktorkan bilangan prima yang relatif lebih besar. Pemfaktoran dilakukan untuk memperoleh kunci privat. Selama bilangan pemfaktoran prima yang besar besar belum ditemukan algoritma yang berhasil memecahkan, maka selama itu pula algoritma rsa akan tetap terjamin keaamanannya [4].

Dengan begitu pesatnya perkembangan era modrenisasi, pemecahan masalah dalam penemuan kode/pembajakan data dapat dilakukan dengan berbagai cara dan bisa juga menggunakan beberapa model algoritma. Diantara jenis algoritma untuk pembajakan data bisa menggunakan algoritma brute force, dimana algoritma ini dapat memecahkan masalah dengan sangat sederhana dalam pembajakan dan pencarian kode dengan cara yang jelas dan lempang [5].

Banyak teknik kriptografi yang telah dipergunakan untuk menjaga keamanan data saat ini, contohnya seperti LOKI, GOST, Blowfish, Vigenere, MD2, MD4, RSA dan lain sebagainya. Masing-masing teknik kriptografi tersebut memiliki kelemahan dan kelebihan [6]. Dalam Penyelesaian permasalahan kode cracking dengan menggunakan algoritma brute force akan menempatkan dan mencari semua kemungkinan kode dengan masukan karakter dan panjang kode tertentu tentunya dengan banyak sekali kombinasi kode [7].

Menjaga keamanan dan kerahasiaan data merupakan hal yang sangat penting untuk melakukan sebuah proses pengiriman data, baik itu data teks dan video melalui jaringan internet yang sudah terkoneksi dengan sangat luas [8]. Didalam pengamanan suatu berkas dokumen, sangatlah dibutuhkan menggunakan sebuah metode/algoritma ilmu kriptografi, agar berkas dokumen tersebut bisa terjamin keamanannya. Dengan

meningkatkan sistem keamanan dari berkas dokumen dapat membantu mengamankan data-data yang ada didalam berkas dokumen pada saat proses pengiriman data, sehingga berkas dokumen tersebut dapat dengan selamat sampai kepada si penerima/yang memiliki hak untuk membuka berkas dokumen tersebut [9]

Kriptografi (*Cryptography*) adalah cabang ilmu matematika tentang persandian untuk menjaga keamanan data. Sistem Kriptografi (*Cryptography System*) atau *Cryptosystem* adalah suatu fasilitas untuk mengkonversikan *plaintext* ke *ciphertext* dan sebaliknya. *Plaintext* adalah data asli, data yang masih bisa dibaca dan dimengerti. Sedangkan *ciphertext* adalah data yang tidak bisa dibaca maupun dimengerti [10].

Dalam dunia kriptografi, kunci untuk mengenkripsi dan mendekripsi suatu pesan rahasia adalah satu elemen terpenting. Kuncilah yang menentukan sebuah *chipertext* dapat dibaca atau tidak. Kerahasiaan kunci justru menjadi hal yang bisa lebih krusial dan penting daripada kerahasiaan *chipertext* itu sendiri, dalam artian pesan (*chipertext* boleh bocor tetapi kunci tidak boleh bocor). Berbagai cara dilakukan untuk menjaga kerahasiaan kunci. Teknik atau algoritma kriptografi kunci public merupakan satu cara yang dikembangkan untuk mengatasi hal tersebut. (kerahasiaan kunci). Kriptografi kunci publik mensyaratkan ada dua buah kunci, yaitu kunci publik yang diinformasikan secara bebas dan digunakan tanpa kerahasiaan, serta kunci privat yang hanya digunakan secara khusus oleh satu orang dan tidak pernah diinformasikan kepada siapapun, sehingga kerahasiaannya sangat terjaga. Walaupun algoritma kriptografi kunci publik popular karena hal tersebut, algoritma kriptografi kunci simetri pun masih banyak digunakan karena lebih mudah penggunaannya, namun perlu dipikirkan bagaimana cara menjaga kerahasiaan kuncinya [11].

BISS (*Basic Interoperable Scrambling System*) merupakan sistem enkripsi dasar video yang dikembangkan oleh *European Broadcasting Union* dan konsorium produsen *hardware*. Awalnya BISS digunakan untuk melindungi hak siar sebuah siaran atau video.BISS adalah *system video scramble* yang paling lemah diantara *system scramble* yang ada. Setiap jenis system acakan memiliki Ca ID 2600 (hexadecimal). Mungkin pernah diminta untuk memasukan Ca Id oleh merek receiver parabola tertentu saat mau memasukan kode BISS untuk membuka sistem acakan, jika yakin system acakan adalah Sistem BISS, maka masukan Ca Id dengan angka 2600.

Karena sistem *BISS* menawarkan mekanisme yang relatif sederhana untuk berebut isi kunci, sehingga dapat memeberikan hak kepada pemegang kendali untuk mendekripsi siaran video tertentu. Hal ini telah banyak digunakan untuk mengamankan data video dengan melakukan pengacakan DSNG (Digital Satellite Gathering) sebuah siaran [12].

III. PEMBAHASAN

Pengambilan data dilakukan dengan cara melakukan perekaman data dari beberapa jenis siaran yang terdeteksi menggunakan jenis acakan BISS dengan menggunakan *Digital Receiver Matrix Prolink HD Eithernet New Youtube* untuk dijadikan sampel data. Proses perekaman data dilakukan dengan menggunakan media *Flash Disk* yang koneksikan ke *Digital Receiver Matrix Prolink HD Eithernet New Youtube*.

Gbr 4. Proses Perekaman Data

Setelah melakukan perekaman data selesai, selanjutnya mencari jenis siaran video yang terdeteksi menggunakan jenis acakan *BISS*.

Gbr 5. Sampel Data Yang Terdeteksi Acakan *BISS*

Berikut sampel data dari hasil perekaman data siaran video yang terdeteksi menggunakan acakan *BISS*:

TABEL I
DATA YANG TERDETEKSI MENGGUNAKAN ACAKAN BISS

No	Freq	Service Id	Key
1	3765	2	24081945622004400
2	3463	1	1000001012345090
3	3880	1617	31fc522aad337b4
4	3880	1615	a788f928460465af
5	3440	906	940d6b0c03a4ea91
6	3880	1618	723923ce5ed9e920
7	3880	1619	410c5aa7ab71607c
8	3545	0	0240a1e32563b73f

9	3764	0	2468110011975300
10	4086	1	2a2bf5fa53d35177
11	4090	4	71d3a9aac77161eaa
12	4105	1	bfcead3a00021113
13	4106	1	bfcead3400021113
14	4148	1	a3434228387032de
15	3667	1	660489f313eec9ca

Dari hasil kunci data *BISS* pada tabel 1. dijadikan sebagai sampel data 16 digit karakter bilangan hexa yang dibutuhkan untuk di *generate* ulang menggunakan fungsi algoritma RSA. Berikut struktur algoritma yang digunakan untuk proses enkripsi ulang :

1. Menentukan Kunci Algoritma RSA.
   Untuk mencari nilai modulus, dibutuhkan 16 (enam belas) digit karakter hexa yang sudah didapat dari data *BISS* yang tertera pada tabel 5.1.
   Input : 16 (enam belas) digit karakterhexa *key* data *BISS*
   Proses : *p*dan*q*bilangan prima yang ditentukan secara *random* (acak)
   $n = p*q$
   dimana*n* merupakan nilai modulus, maka $\phi(n) = (p-1) * (q-1)$
   Output : n / $\phi(n)$

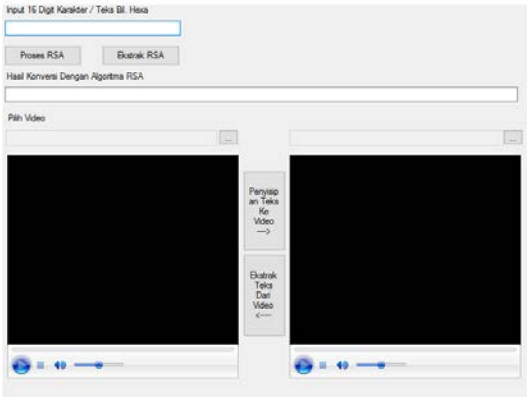
   Sampel key data *BISS* yang dijadikan sebagai *input* data adalah *4332e358d948c9*, selanjutnya key data *BISS* tersebut akan dienkripsi ulang dengan menggunakan algoritma RSA. Untuk nilai *p* dan *q* adalah bilangan prima yang ditentukan secara *random* (acak). Maka hasil dari sampel key data *BISS* yang sudah dienkripsi diatas adalah " KuaOAoAgBut23JQ02u2mVSZJdj1RoQ0ReAZ + M3cl9UIrQOYNljG0BFvPb2M + 7gNo0nqveh14P/+ p7zvqp3QFZJyUDKQ9tSCYaffIWVd9pYRqC1t/2Ift9l7OYJeWx9lSfZQq3B5WFM7KF7GKG0NmiVgoQVAj0Sn + dWG4/78iDTo= "

2. Penyisipan Karakter Hexa yang sudah di enkripsi dengan algoritma RSA.
   Untuk penyisipan karakter hexa dilakukan dengan cara menentukan jarak *frame* untuk video secara acak yang akan dipotong untuk disisipkan bagian-bagian karakter hexa tersebut. Video yang sudah dipilih, dipotong berdasarkan bagian per bagian untuk disisipkan karakter hexa dengan menggunakan karakter *array* dan ukuran *byte* dalam pemotongan yang nantinya digunakan untuk pembentukan ukuran *byte* yang baru dari video dan penyisipan karakter hexa.
   Input : n / $\phi(n)$
   Proses : *byte*[*make new size*] = new *byte*[*filebyte.length* + *encryptedbyte.length* + *datalength.length*]
   //penyisipankarakter kebagian video yang baru
   Array.copy(*datalength*, 0, x*_byte*, 0, *datalength.length*)
   Array.copy(*encryptedbyte*, 0, *x_byte*, *datalength.length*, *encryptedbyte.length*)
   Array.copy(*filebyte*, 0, *x_byte*, *datalength.length* + *encryptedbyte.length*, *filebyte.length*)
   Output : File Video

Hasil dari key data *BISS* yang sudah di enkripsi dengan algoritma RSA yaitu " KuaOAoAgBut23JQ02u2mVSZJdj1RoQ0ReAZ + M3cl9UIrQOYNljG0BFvPb2M + 7gNo0nqveh14P/+ p7zvqp3QFZJyUDKQ9tSCYaffIWVd9pYRqC1t/2Ift 9l7OYJeWx9lSfZQq3B5WFM7KF7GKG0NmiVgoQ VAj0Sn + dWG4/78iDTo= " akan diadikan sebagai *input* (masukan) yang akan selanjutnya akan disisipkan kedalam file video. Selanjutnya proses pemotongan bagian-bagian *frame* video yang akan disisipkan karakter key data *BISS* yang sudah di enkripsi.

IV. HASIL

Setelah dilakukan proses-proses yang telah dilakukan mulai dari pengumpulan data, menganalisa model data, studi literatur, uji coba proses enkripsi dan uji coba proses dekripsi, tahap selanjutnya adalah perancangan aplikasi yang akan digunakan untuk mempercepat proses. Berikut adalah rancangan dari tampilan tatap muka yang akan digunakan untuk proses pengamanan data *BISS*.

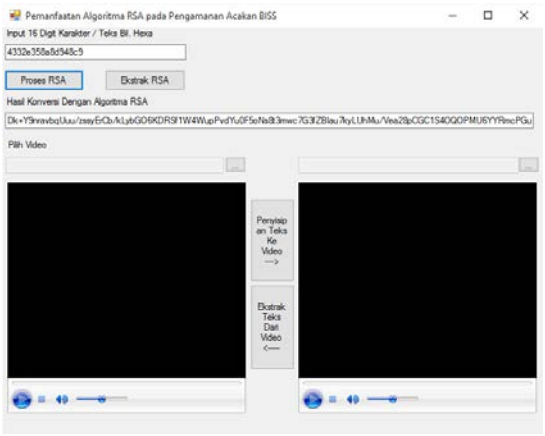


Gbr 6.Tampilan Antar Muka Aplikasi Pengamanan Acakan BISS

Pada gambar 6. merupakan tampilan dari rancangan aplikasi yang digunakan untuk meningkatkan keamanan sebuah data acakan *BISS* dan video sebelum data video dikirimkan kepada penerima/pembeli siaran video (hak ekslusif).

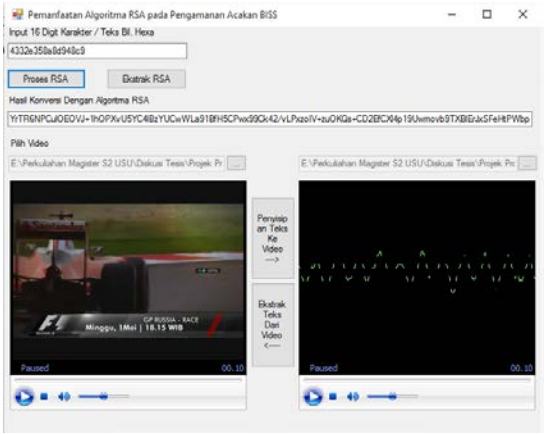
Untuk proses enkripsi awalnya adalah dengan menyiapkan karakter hexa *BISS* berjumlah 16 (enam

belas) karakter yang telah direkam dengan menggunakan *Receiver Matrix Prolink HD Eithernet New Youtube*. Sebagai sampel data yang sudah didapat adalah terdapat pada tabel 1.



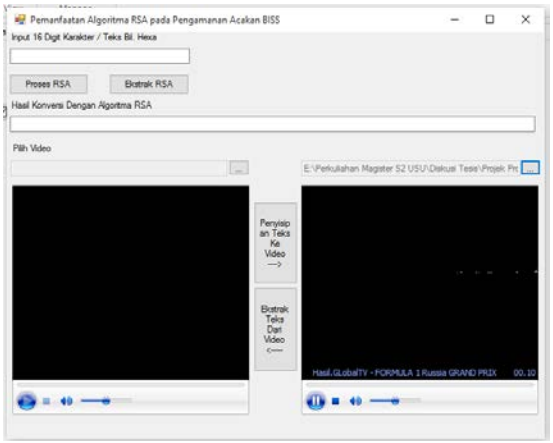
Gbr 7. Input Karakter Hexa Dan Proses Enkripsi

Langkah selanjutnya adalah pemilihan video, penyisipan karakter hexa dan pengacakan file siaran video. Setelah proses pengenkripsian 16 digit karakter hexa key data *BISS* dengan menggunakan algoritma RSA selesai maka akan didapatkan hasil dari karakter hexa yang baru, jumlah karakter yang tadinya masih 16 (enam belas) digit berubah menjadi lebih dari 16 (enam belas) digit, itu dikarenakan hasil dari proses enkripsi yang telah dilakukan dengan menggunakan algoritma RSA, selanjutnya proses pemilihan video dari hasil rekaman melalui media *Reciever Matrix Prolink HD Eithernet New Youtube*, sampel data file videonya adalah "GLobalTV - FORMULA 1 Russia GRAND PRIX" yang akan di enkripsi ulang dengan menyisipkan karakter hexa yang sudah di enkripsi dengan algoritma RSA. Berikut akan ditampilkan pada gambar 8.



Gbr 8. Proses Penyisipan Karakter Hexa Dan Pengacakan Video

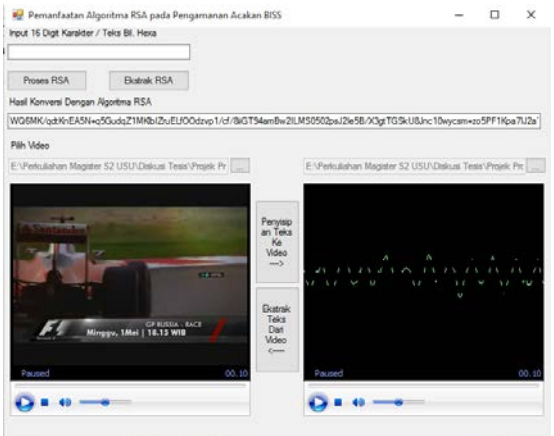
Untuk proses dekripsi, file yang sudah dienkripsi akan didekripsi kembali untuk menghilangkan acakan dan mengembalikan karakter hexa, sehingga file video tersebut dapat diputar kembali oleh sipenerima video. Selanjutnya melakukan pemilihan file video yang sudah dienkripsi dengan menggunakan aplikasi yang sudah dirancang.



Gbr 9. Pemilihan Video Yang Masih Dienkripsi

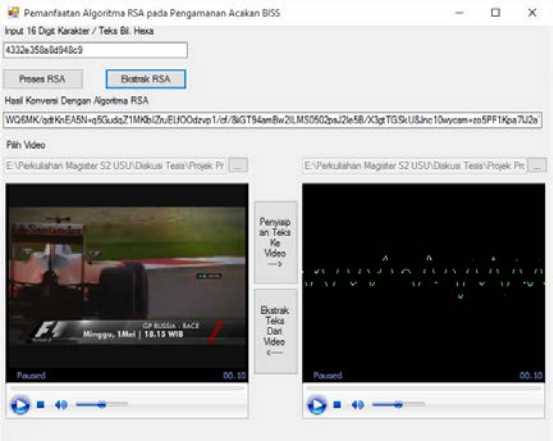
Setelah memilih file video yang sudah dienkripsi, file video akan masuk ke media pemutar pada sisi bagian kanan, file video terlihat masih teracak dan tidak dapat diputar secara normal, sedangkan pada pemutar video disebelah kiri masih gelap, karena proses ekstrak atau pemisahan karakter hexa dari video belum dilakukan.

Selanjutnya melakukan proses ekstrak/pemisahan karakter hexa dari file video dengan mengklik tombol ekstrak. Maka akan terlihat file video akan berpindah kebagian pemutar video yang ada disebelah kiri dan karakter hexa pun terpisahdarifile video.



Gbr 10. Pemisahan Karakter Hexa Dari File Video

Selanjutnya dilakukan proses dekripsi algoritma RSA ke karakter hexa *BISS*. Pada proses ini karakter hexa yang sudah dipisahkan dari file video masih terenkripsi dengan fungsi dari algoritma RSA, akan dikembalikan kedalam bentuk semula, yaitu karakter hexa *BISS*.



Gbr 11. Proses Ekstrak Karakter Dari RSA Ke *BISS*

Berikut tabel testing yang diperoleh dari pengujian pada aplikasi yang dibuat oleh penulis.

TABEL II
HASIL TESTING APLIKASI

No	Objek	Halaman/Fungsi yang dituju	Hasil
1	Text	Menyisipkan karakter Hexa	Berhasil
2	Tombol Proses RSA	Melakukan proses enkripsi RSA	Berhasil
3	Text	Menampilkan hasil enkripsi RSA	Berhasil
4	Tombol Cari Video Normal	Membuka file video yang ingin di enkripsi	Berhasil
5	Video Dialog	Pemutar file video yang masih normal	Berhasil
6	Tombol Penyisipan Teks ke Video	Melakukan proses penyisipan karakter hasil enkripsi dan proses enkripsi video	Berhasil
7	Video Dialog	Pemutar file video yang sudah dienkripsi	Berhasil
8	Tombol Cari Video Terenkripsi	Membuka file video yang sudah terenkripsi	Berhasil
9	Tombol Ekstrak Teks dari Video	Proses pemisahan karakter dari video	Berhasil
10	Tombol Ekstrak RSA	Proses dekripsi karakter RSA ke 16 digit karakter Hexa	Berhasil

Berikut tabel hasil dari enkripsi karakter hexa key data *BISS* dengan algoritma RSA.

TABEL III
SAMPEL DATA ENKRIPSI KARAKTER *BISS* DENGAN ALGORITMA RSA

No	Freq	Key Data BISS	Enkripsi RSA	Hasil
1	3765	2408194562200440	UhPIqE/HqUev5gDbOJ94ZZRCQ9F17J82GHTP5ewSk2IFW72i7zGFfttMXiDHVs7uprS+jE+TntkqWL81hUpdmcER1o52ynTTYG9fnQffrEHIPm/ZtbLNHYhbzNE0EfOXxxDFzWAhZDNbZEf01yWmVqlmsIZZ+ED+ikrqqfwu0ME=	Berhasil
2	3463	10000	nPzqptqA8obJ23GlHZwvf	Berhasil
		0101234509	XOzSG4X9zUujOmhHK7Xqdd7ZB0AKdN92yPIrcjvC1E12wy8lBimCXcBxIH8CSha5Ke62Or3MB/Yg3fwkYJgIdl/TKtKoHt3IK/zTmcIkrc867O5+yBwbKO5FDHgMmT3Op5A7jEI0aL3j/mmFXTsf3Q=	
3	3880	31fc522aad337b4	z84h/jC7ebprDTt9NOkaeMDWJuJ8vHhtAMhjuk5+opqMjAt5bfuubJjYl0HpXk7GVmX2X202zdEw4eIOWF6kXNaz4g7Qg2ET9X8AbuKGFhwqw/sC0v0Q2y5zKBdOJM22Tf0B/29hYJPPi1swmFIEiOG/pix1SvC6aTIqvI4/JHE=	Berhasil
4	3880	a788f928460465af	GJnNEBexBlOl6Bmv0jc6VDg8CTaZxyOj32sDt/JqjizpDCHROwaisog3q8FhsLWX6Ij5qZFi/KmRYMXejyLj742HC+/nY7yDpLxelwRok9wUCuOn9PnwooKpMQta/lzSM60XoXjU3Ol3jBU/XX/ZkKS3DuErIrWYW5hx5T0Q8C0=	Berhasil
5	3440	940d6b0c03a4ea91	Fgs6UwyxrshmbfSpWNkrCx1bCvZuGXqSfes1DnsuLBvsvh44NLVm7DFkN0gt43QZIeF4xTosybRXUIz1pCa/+/MAgTMvbmFZ29F5ew+OuX87PsO5QJGR/iPBDTpa9b2rDH2JQbuVXjry057cqwonR8AVrLfeIdbf4IWTnWsaH2E=	Berhasil

V. SIMPULAN

Berdasarkan hasil pembahasan diatas, maka diperoleh suatu model yang baru, yang dapat meningkatkan sistem keamanan data dari acakan BISS sebuah data siaran video dengan menggunakan fungsi algoritma RSA. Dari hasil pengujian aplikasi menggunakan fungsi algoritma RSA, dapat memberikan masukan-masukan data secara acak dan otomatis serta menghasilkan daftar nilai kunci yang juga secara acak.

UCAPAN TERIMAKASIH

Ucapan terimakasih kami sampaikan kepada Direktorat Riset dan Pengabdian Masyarakat Direktorat JenderalPenguatan Riset, Teknologi dan Pendidikan Tinggi atas Pendanaan Penelitian Dosen Pemula (PDP) Tahun Pelaksanaan 2018.

REFERENSI

[1] Gani, D. *Jenis Acakan Dengan System BissKey*. 2013. http://mahasiswa.ung.ac.id/521413035/hone/2013/9/page/39. Diakses pada tanggal 9 Juni 2018.

[2] Gunawan, I. *Pengamanan Acakan BISS Menggunakan Algoritma RSA*. Jurnal Riset Informasi dan Teknik Informatika (JURASIK), Vol. 2. No. 1. pp. 58-63. Juli 2017.

[3] Gani, P. I. & Abdurrohman, M. *Selective Encryption of Video MPEG use RSA Algorithm*.1st International Conference on Information Technology, Computer and Electrical

Engineering (ICITACEE). IEEE 978-1-4799-6432-1/14/$31.00. 2014.

[4] Gunawan, I. *Kombinasi Algoritma Caesar Cipher dan Algoritma RSA untuk Pengamanan File Dokumen dan Pesan Teks*. Jurnal Nasional Informatika dan Teknologi Jaringan (InfoTekjar), Vol. 2, No. 2, pp. 27-32. Maret 2018.

[5] Gunawan, I. *Penggunaan Brute Force Attack dalam Penerapannya pada Crypt8 dan CSA-Raibow Tool untuk Mencari BISS*. Jurnal Nasional Informatika dan Teknologi Jaringan (InfoTekjar), Vol. 1, No. 1, pp. 52-55. September 2016.

[6] Gunawan, I., Sumarno, Irawan, E., Tambunan, H. S. *Pengamanan Berkas Dokumen Menggunakan Fungsi Algoritma Steganografi LSB*. Jurnal Ilmu Komputer dan Informatika (ALGORITMA). Vol. 2, No. 1, pp. 61-65. April 2018.

[7] Sianipar, K. D. R., Purba, L. C., Siahaan, S. W., Gunawan, I., Sumarno. *Pengamanan File Gambar Menggunakan Fungsi Algoritma Steganografi LSB dari Serangan Brute Force*. Jurnal Teknik Informatika (TECHSI). Vol. 10, No. 1, pp. 155-162. April 2018.

[8] Gunawan, I. & Sumarno. *Penggunaan Algoritma Kriptografi Steganografi Least Significant Bit untuk Pengamanan Pesan Teks dan Data Video*. Jurnal Sains Komputer & Informatika (J-SAKTI). Vol. 2, No. 1, pp. 57-63. Maret 2018.

[9] Gunawan, I., Sumarno, Tambunan, H. S., Irawan, E., Kirana, I. O. *Fungsi Algoritma Kriptografi Hill Cipher untuk Pengamanan File Gambar dan Pesan Teks*. Jurnal Teknik Informatika (TECHSI). Vol 10, No. 1, pp. 119-128. April 2018.

[10] Munawan. *Perancangan Algoritma Sistem Keamanan Data Menggunakan Metode Kriptografi Asimetris.* Jurnal Komputer dan Informatika (KOMPUTA). Edisi 1, Vol. 1. 2012

[11] Wahyuni, A. *Keamanan Pertukaran Kunci Kriptografi dengan Algoritma Hybrid : Diffie-Hellaman dan RSA.* Majalah Ilmiah Informatika, Vol. 2, No. 2. 2011.

[12] Priyono, B. S. *BISS vs CSA – Istilah-istilah Seputar Persatelitan*.http://www.forumsatelit.com/tutorial-teknis-parabola-dan-televisi-satelit/istilah-istilah-seputar-dunia-persatelitan/40/.Diakses pada tanggal 9 Juni 2018.