PENERAPAN *V.S.N HARDWARE KEY SCHEME* DENGAN *RSA CRYPTOSYSTEM* UNTUK PENGAMANAN PERANGKAT LUNAK

[1]Andy Victor, [2]Taufan Maulana Putra
[1, 2]Program Studi Teknik Teknik Informatika, Sekolah Tinggi Manajemen Informatika dan Komputer LPKIA
[1]abang@lpkia.ac.id, [2]jcyberknight@gmail.com

Abstrak—Jumlah pencurian data perusahaan meningkat dari tahun ke tahun. Hal ini dikarenakan kurangnya kesadaran pihak perusahaan akan pentingnya suatu sistem keamanan yang efisien dan unik. Pihak perusahaan mengira bahwa hanya dengan password, data tersebut sudah sangat aman. Pada penelitian ini, dibuat suatu skema keamanan data, yakni V.S.N Hardware Key dengan algoritma kriptografi RSA. Volume Serial Number yang terdapat di dalam suatu hardware yang berbentuk bilangan hexadecimal diambil dengan fungsi API GetVolumeInformation dan diubah menjadi bilangan decimal yang kemudian digunakan sebagai otentifikasi dalam pengaksesan sebuah aplikasi. RSA merupakan salah satu algoritma kriptografi asimetris yang menggunakan sepasang kunci, yaitu kunci publik dan kunci privat. Panjang kunci dapat diatur, dimana semakin panjang bit pembentukan kunci maka semakin sukar untuk dipecahkan karena sulitnya memfaktorkan dua bilangan yang sangat besar. Aplikasi yang dirancang dalam mendukung V.S.N Hardware Key dengan algoritma kriptografi RSA telah diuji dengan melibatkan data-data confidential perusahaan, dan terbukti ampuh dalam membuat para intruder gagal melakukan pencurian data penting perusahaan termasuk penggandaan ilegal. Secara umum V.S.N Hardware Key dengan algoritma kriptografi RSA digunakan sebagai skema penyempurna pengamanan data perusahaan yang secara umum berbentuk aplikasi sistem informasi.

Kata kunci: *V.S.N hardware key*, *volume serial number*, *get volume information*, keamanan data, *RSA*

I. PENDAHULUAN

Melihat dari berbagai persitiwa pencurian data maupun pihak-pihak yang dengan sengaja menyalin data yang bersifat confidential untuk tujuan tertentu yang sering terjadi di Indonesia, maka diperlukan suatu teknologi pengamanan data yang baru dan unik sehingga dapat membingungkan para peretas. Tujuan dari keamanan data adalah untuk menemukan cara-cara untuk mencegah dari exploitasi demi menjaga keutuhan 3 aspek utama yakni, *Confidentiality, Integrity dan Availability* [1].

Metode / Teknik yang umum digunakan untuk pengamanan data seperti *MD5 Scheme* (Ron Rivest, 1992), *RSA Encryption Scheme* (Rivest, Shamir, Adleman, 1977) dan *Dongle* (Pete Dowson, 1980). Setelah melakukan studi perbandingan terhadap metode-metode keamanan diatas, MD5 memiliki kelebihan, yaitu dapat digunakan sebagai *Intruder Detection System* yang mengambil nilai hash dari sebuah file, namun MD5 memiliki kelemahan yakni, *Collision*, maksudnya ada 2 atau lebih teks yang menghasilkan nilai hash yang sama [2]. RSA memiliki kelebihan yaitu, kesulitan untuk memfaktorkan bilangan besar modulus n menjadi faktor-faktor primanya, namun RSA memiliki kelemahan yakni, class exponent weak formula [3].

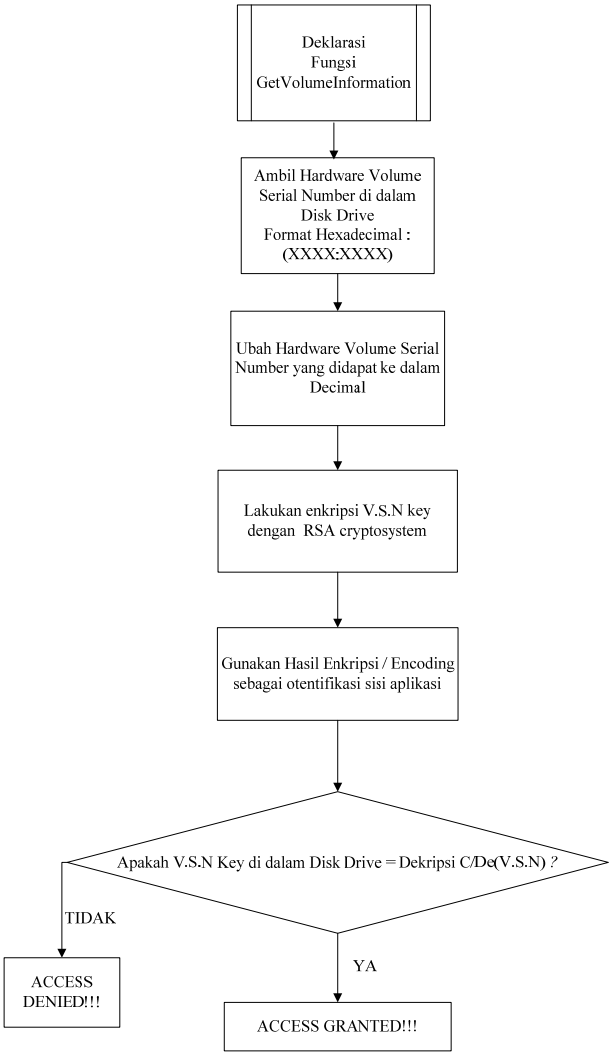
Proteksi Dongle dapat menjadi jalan keluar dalam memberikan proteksi yang aman, karena sistem enkripsi yang terhubung antara sisi hardware yang ditanamkan pada *micro chip* dengan sisi software yang dapat meminimalisir penyadapan saluran komunikasi data, namun Dongle memiliki kelemahan pada kurangnya efisiensi algoritma dan efisiensi biaya.

Algoritma *V.S.N Hardware Key* adalah solusi pengembangan baru yang terbukti efektif untuk mengamankan data secara efisien, dengan menggabungkan konsep Dongle dan penerapan *Volume Serial Number* [4] yang digunakan sebagai *Hardware Key* untuk acuan otentifikasi. Dengan mengimplementasikan algoritma *V.S.N Hardware Key*, maka keunggulannya adalah storage device yang digunakan sebagai kunci dapat bersifat portable sehingga memudahkan kita untuk membawa kemanapun hardware key dan yang paling utama bahwa pembuatan skema proteksi dengan *V.S.N Hardware Key* ini didasari pada konsep pengalihan persepsi Houdini, membuat orang percaya melalui penglihatan, pendengaran dan perasaan sehingga orang mengira itu adalah kenyataan yang sebenarnya.

II. METODE PENELITIAN

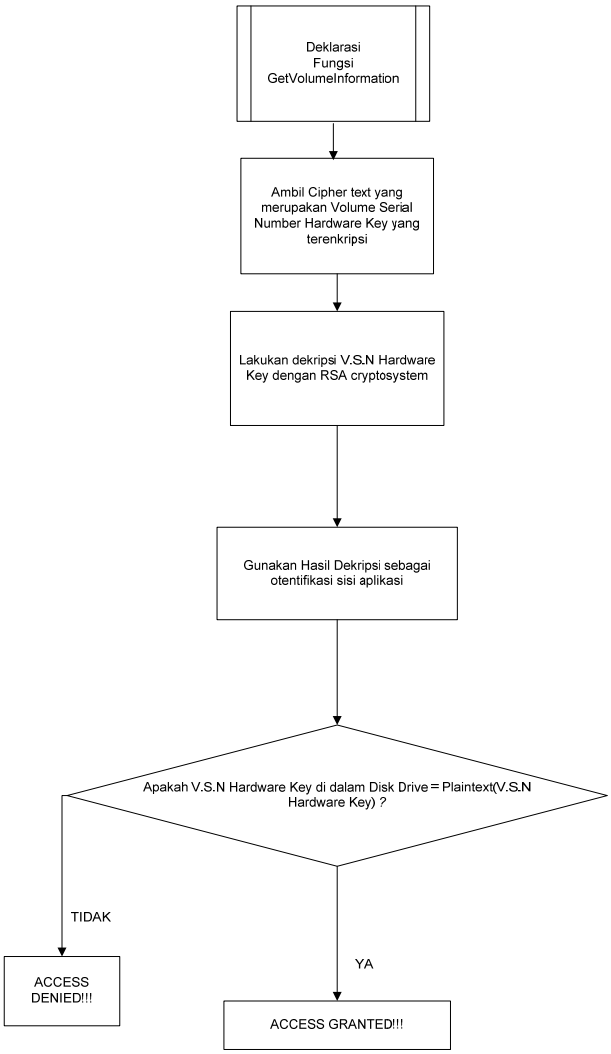
Penelitian ini berkaitan dengan proses penyandian kunci V.S.N meliputi proses enkripsi dan proses dekripsi menggunakan algoritma kriptografi RSA. Hal yang melatarbelakangi peneliti untuk menambahkan algoritma kriptografi dikarenakan ingin mengetahui efektivitas dari segi kompleksitas kerumitan proses enkripsi maupun

enkripsi yang digunakan untuk tetap menjaga volume serial number key. Peneliti menggunakan eksperimental kuantitatif, dimana akan ada 2 kelompok, yakni pengenkripsian serta pendekripsian kunci V.S.N dengan menggunakan algoritma kriptografi RSA. Pada Gambar 1 akan dijelaskan cara kerja / mekanisme dari Pengenkripsian V.S.N Hardware Key Scheme.



Gambar 1 Flowchart enkripsi v.s.n hardware key

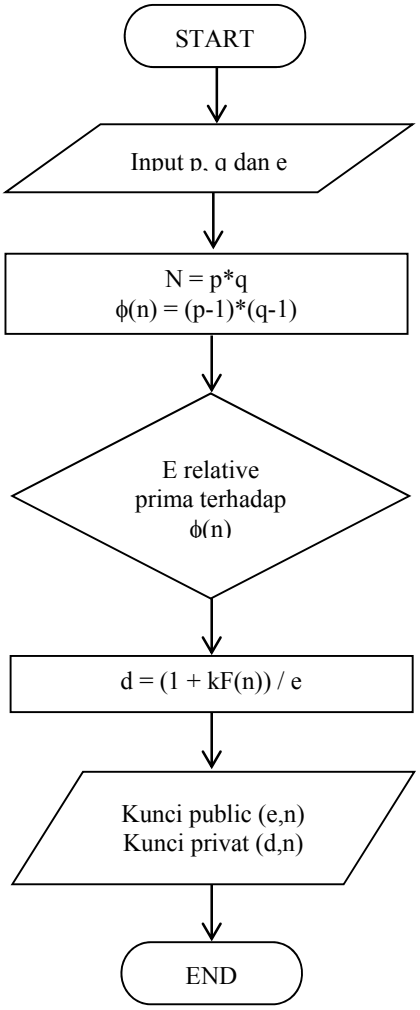
Berdasarkan Gambar 1 tersebut diatas, pertama, sistem akan mengambil fungsi library API, yakni GetVolumeInformation(), yang mana digunakan untuk mengambil sebuah hardware key yang bersifat dinamis(bergantung pada format storage drive tersebut berdasarkan tanggal), lalu secara otomatis fungsi GetVolumeInformation() akan mengambil dalam format Hexadecimal 8 digit (dipisahkan oleh -), kemudian kita konversi bilangan hexadecimal ke dalam bilangan decimal dengan maksud & tujuan agar lebih mudah untuk membagi ke dalam blok blok pada saat proses enkripsi menggunakan algoritma kriptografi RSA. Setelah itu lakukan enkripsi pada tiap blok-blok plaintext yang berisi bilangan decimal, dengan rumus $y_i = x_i^{PK} \bmod r$, setelah itu maka hasil enkripsi tersebut dijadikan otentifikasi dari sisi aplikasi. Setelah proses enkripsi dilakukan maka proses berikutnya adalah proses dekripsi seperti pada Gambar 2



Gambar 2 Flowchart dekripsi b.s.n hardware key

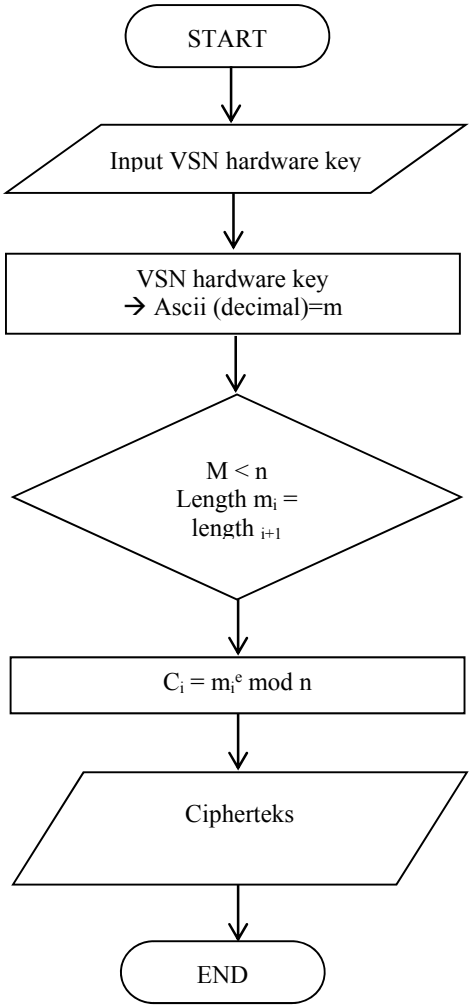
Berdasarkan Gambar 2 tersebut diatas, untuk proses pendekripsiannya sistem tetap akan mengambil terlebih dahulu fungsi GetVolumeInformation(), untuk mendapatkan Hardware Key dan akan digunakan sebagai otentifikasi. Setelah itu sistem akan melakukan dekripsi cipher text yang merupakan hasil enkripsi dari V.S.N Hardware Key. Setelah itu, maka hasil dekripsi yang berupa plaintext nyata, yakni bilangan decimal dari konversi V.S.N Hardware Key yang sebelumnya berbentuk bilangan hexa decimal digunakan untuk otentifikasi dengan hardware yang tertanam pada komputer. Untuk proses pembangkitan kunci algoritma

kriptografi RSA dapat dilihat pada Gambar 3 yaitu pertama pilih dua bilangan prima secara acak. Bilangan itu diberi besaran p dan q. setelah itu hitung nilai n = p.q, lalu hitung juga $\phi(n) = (p-1)(q-1)$. Selanjutnya, pilih kunci publik yang disimbolkan dengan e. Syarat dari pemilihan kunci ini adalah e harus relative prima terhadap $\phi(n)$. Dan lakukan pembangkitan kunci privat dengan persamaan $d = (1+ k\phi(n))/e$. lalu dari hasil pembangkitan sepasang kunci, maka kita akan mendapatkan kunci publik dengan pasangan (e,n) , serta kunci privat dengan pasangan (d,n).



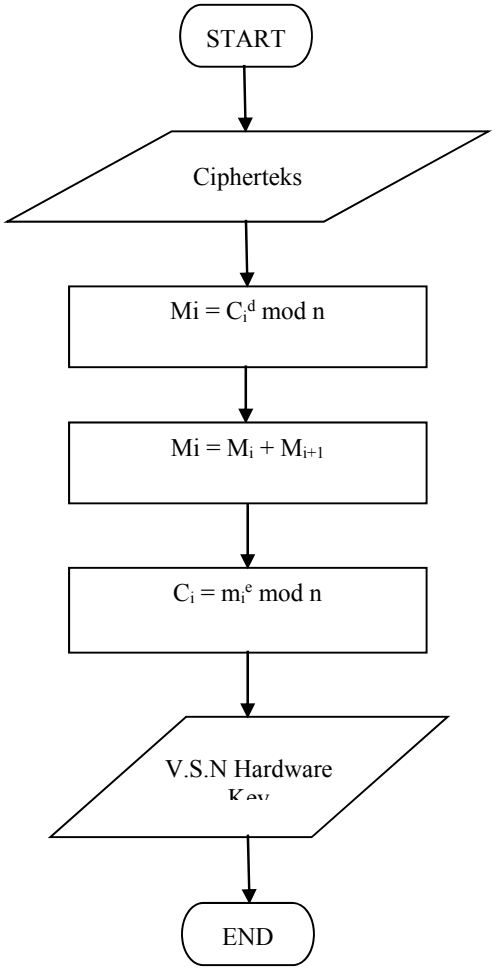
Gambar 3 Flowchart pembangkitan kunci rsa

Dari Gambar 3 tersebut diatas, untuk proses enkripsi langkah pertama yang harus dilakukan ialah mengambil nilai e dan n dari proses pembangkitan kunci, kemudian masukan teks yang akan dienkripsi (dalam hal ini V.S.N Hardware Key), berkas yang akan dienkripsi di ubah kedalam bentuk decimal sesuai dengan tabel ASCII. Selanjutnya membagi plainteks tersebut menjadi beberapa blok (mi), dengan syarat mi < n dan panjang(mi) = panjang(mi+1). Setelah itu setiap blok dari berkas tersebut dienkripsi menggunakan pasangan kunci publik(e,n). Proses enkripsi ini dapat dilihat pada Gambar 4



Gambar 4 algoritma enkripsi rsa

Untuk proses dekripsi, dapatkan cipherteksnya, dengan begitu informasi tersebut tidak akan dapat dibaca lagi oleh orang tanpa melalui proses dekripsi. Proses dekripsi pada algoritma RSA ini memerlukan yang dinamakan dengan kunci privat. Kunci privat hanya diketahui oleh orang yang berhak atas informasi tersebut. Proses dekripsi dapat dilihat pada Gambar 5.



Gambar 5 algoritma dekrispi rsa

A. Prosedur penelitian

Prosedur yang dilakukan dalam penelitian ini adalah sebagai berikut :

1. Sebuah flash drive yang sudah akan diambil dalam format volume serial number.
2. Dalam proses penerapan V.S.N Hardware Key Scheme dengan algoritma kriptografi RSA, kunci V.S.N Hardware Key yang sudah dikonversi ke dalam bilangan decimal akan dijadikan sebagai plaintext yang kemudian dienkripsi.
3. V.S.N Hardware Key yang sudah berupa ciphertext kemudian ditanam ke dalam perangkat lunak di sisi otentifikasi pada login form.
4. Dilakukan verifikasi terhadap username, password dan security key yang merupakan V.S.N Hardware Key.
5. Pada saat proses verifikasi V.S.N Hardware Key yang ditanam di dalam perangkat lunak dengan kunci V.S.N Hardware Key yang terdapat di dalam perangkat keras, dilakukan proses dekripsi terlebih dahulu terhadap V.S.N Hardware Key yang ditanam guna melindungi dari para penyadap.

B. Variabel penelitian

Beberapa variabel yang terkait dalam penelitian ini adalah:

1. Tipe flash drive yang digunakan.
2. Volume Serial Number dari flash drive dalam satuan hexadecimal.
3. Jenis File System yang masuk ke dalam lingkup sistem operasi Windows.

C. Populasi & sampel

Populasi dari penelitian ini adalah sekumpulan kunci kunci V.S.N yang peneliti ambil secara random dari beberapa hard disk maupun usb flash drive. Struktur kunci V.S.N terdiri dari 8 digit dan merupakan bilangan basis 16 (hexadecimal). Maka dari itu untuk memudahkan pengelolaan populasi maka peneliti melakukan translasi ke bilangan basis 10 (decimal). Lalu untuk sampling diambil 3 kunci V.S.N. yang mana untuk medapatkan kunci V.S.N yang sudah dikonversi ke dalam bilangan decimal. Berikut ini peneliti mengambil sample dari 3 item yang memiliki file system yang berbeda, dapat dilihat pada tabel I dibawah ini :

TABEL I
SAMPEL FILE SYSTEM BESERTA V.S.N HARDWARE KEY

No	Item Sample	File System	V.S.N Hardware Key
1	Flash Drive Transcend 16GB	NTFS	2823-CF89
2	Flash Drive Kingston 1GB	FAT32	DEC9-1E6D
3	Flash Drive Sony 2GB	FAT	1E98-27TF

III. HASIL DAN PEMBAHASAN

Pada penelitian ini, penulis menerapkan algoritma V.S.N terhadap perangkat lunak Sistem Informasi Kepegawaian PT. TVRI (Persero) yang dibuat menggunakan *compiler Microsoft Visual Basic.NET versi 9.0*.

IV. *Screenshoot* Perangkat Lunak Penelitian

Hasil implementasi dari penerapan algoritman V.S.N Hardware Key pada proses login atau otentifikasi perangkat lunak Sistem Informasi Kepegawaian dapat dilihat pada. Pertama user mengisi user dan access code, jika flash drive terhubung kekomputer maka status security key secara otomatis akan berubah menjadi "Hardware Key Is Actived) seperti pada Gambar 6. Ketika user menekan tombol enter maka form access granted akan muncul ketika value user, access code dan security key bernilai true seperti pada Gambar 7. Apabila bernilai false maka access denied seperti pada Gambar 8

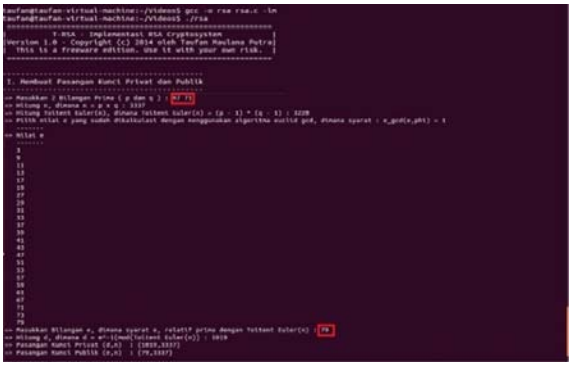
Gambar 6 V.s.n hardware key bernilai true

Gambar 7 V.s.n hardware key bernilai true

Gambar 8 V.s.n hardware key bernilai false

V. Pembangkitan pasangan kunci publik & privat

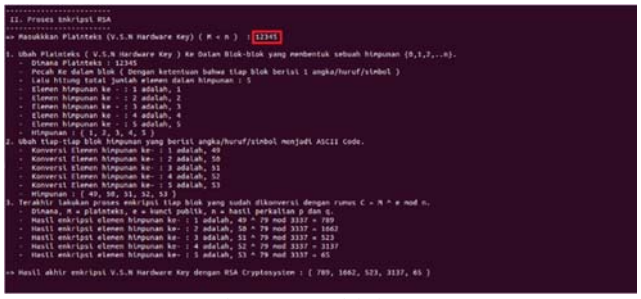
Untuk pembangkitan pasangan kunci seperti pada gamabar 9, pertama-tama user mengisi value p dan q, yang mana merupakan bilangan prima, Misal : 47 71. Kemudian secara otomatis akan dikalkulasi nilai n, yang mana hasil perkalian value p dan q , hasil : 3337. Secara otomatis pula user akan ditampilkan nilai e guna mempermudah user dalam menentukan nilai e. Nilai e digunakan sebagai kunci public yang ditentukan dengan algoritma Euclid GCD (Greatest Common Divisor). Dan nilai d pun akan didapatkan yang mana merupakan kunci privat dengan persamaan : d = e ^ -1 mod (toitent euler(n)). Pada proses terakhir, terbentuklah sepasang kunci publik dan kunci privat.



Gambar 9 Pembangkitan pasangan kunci publik & private

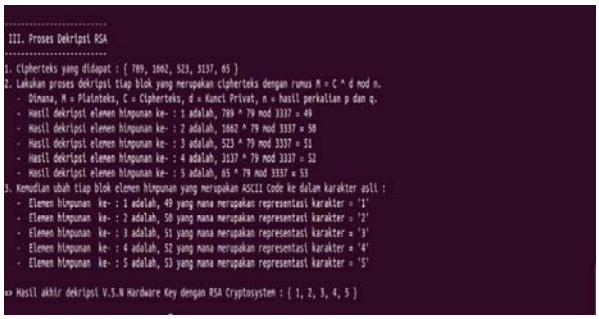
VI. Enkripsi dan Dekripsi RSA

Proses Enkripsi RSA Cryptosystem yang ada pada Gambar 10 yaitu User mengisi value plainteks (dalam hal ini ditujukan untuk V.S.N Hardware Key), dengan syarat : M < n. Kemudian secara otomatis plainteks akan dipecah menjadi blok-blok yang mana batasannya adalah, 1, sehingga menghasilkan himpunan : { 1, 2, 3, 4, 5 }. Lalu dari himpunan yang didapat maka tiap elemen-elemen himpunan akan dikonversi ke dalam ASCII Code berdasarkan tabel yang sudah ditetapkan oleh standar ANSI, sehingga menghasilkan himpunan baru : { 49, 50, 51, 52, 53 }. Pada proses terakhir, enkripsi tiap elemen-elemen himpunan baru dengan ketentuan : Ci = Mi ^ e mod n. maka terbentuklah himpunan cipherteks : { 789, 1662, 523, 3137, 65 }.



Gambar 10 Proses enkripsi RSA

Proses Dekripsi RSA Cryptosystem yang ada pada Gambar 11 yaitu secara otomatis cipherteks akan diambil dari hasil proses enkripsi sebelumnya. Kemudian akan dilakukan proses dekripsi tiap elemen-elemen himpunan cipherteks dengan ketentuan : Mi = Ci ^ d mod n. maka terbentuklah himpunan plainteks : { 49, 50, 51, 52, 53 }. Lalu dari himpunan plainteks yang didapat akan dikonversi ke dalam representasi karakter asli, dikarenakan plainteks yang didapat merupakan ASCII Code, sehingga menghasilkan himpunan plainteks akhir :{ 1, 2, 3, 4, 5 }.



Gambar 11 Proses dekripsi RSA

VII. Hasil Pengujian

Hasil dari pengujian dari skema keamanan yang ditanam ke dalam sebuah login form dapat dilihat pada tabel II

TABEL II HASIL PENGUJIAN

Fungsi yang di uji	Cara Pengujian	Pengamatan	Kesimpulan
Validasi Authentication (TRUE)	User mengisi textbox Username : ad''min Password : TVRI1962 Serta user mencolokkan hardware key yang membuat status Security Key : HARDWARE KEY IS ACTIVE GRANTED	Sesuai dengan prediksi, user dapat masuk ke dalam sistem, dikarenakan *value* username, password beserta Hardware Key bernilai TRUE.	Otentifikasi bernilai TRUE, sehingga user dapat masuk ke dalam form ACCESS GRANTED VERIFIED
Validasi Authentication (FALSE)	User mengisi textbox Username : ad''min Password : TVRI1962 namun user tidak mencolokkan hardware key yang membuat status Security Key : UNKNOWN	Sesuai dengan prediksi, bahwa tanpa adanya flash drive yang digunakan sebagai kunci, meskipun username dan password benar, user tidak akan masuk ke dalam sistem akan tetapi masuk ke dalam form ACCESS DENIED	Otentifikasi bernilai FALSE, kemudian masuk ke dalam form ACCESS DENIED (VERIFIED)
Transformasi Plainteks (*V.S.N Hardware Key*) Pada Proses Enkripsi *RSA* & Transformasi Cipherteks menjadi Plainteks (*V.S.N Hardware Key*) Pada Proses Dekripsi RSA	User mengisi *value* p : 47 , q : 71, e : 79, dan memasukkan plainteks (*V.S.N Hardware Key*) : 3737722477, maka secara otomatis akan memproses seluruh mekanisme *RSA Cryptosystem* (Enkripsi & Dekripsi) yang menghasilkan Cipherteks : 523 1773 523 1773 1773 1662 1662 3137 1773 1773 serta mengembalikan Plainteks : 3 7 3 7 7 2 2 4 7 7	Sesuai dengan perhitungan, bahwa terjadinya proses enkripsi dengan menggunakan algoritma *RSA* yang menyebabkan transformasi dari plainteks menjadi Cipherteks yang mana memiliki ukuran lebih panjang.	Panjang(Cipher teks) > Panjang(Plain teks) (VERIFIED)

Dan berikut ini merupakan elemen-elemen pada algoritma kriptografi RSA yang peneliti gunakan pada saat pengujian, yang terdiri dari sepasang kunci publik(e,n) serta kunci privat(d,n) yang digunakan dalam proses enkripsi maupun dekripsi, p dan q serta n (hasil perkalian p dan q) :

1. p = 47 , q = 71, n = 3337, e = 79, d = 1019
2. Pasangan Kunci Publik : { 79, 3337 }
3. Pasangan Kunci Privat : { 1019, 3337 }

IV. KESIMPULAN

Penerapan algoritma V.S.N Hardware Key Scheme dengan algoritma kriptografi RSA dapat digunakan sebagai alternatif dalam proteksi perangkat lunak. Terbukti pada hasil pengujian di perangkat lunak sistem informasi kepegawaian PT. TVRI (PERSERO) banyak intruder yang masuk ke dalam Trap Scheme. Dilain pihak tidak dapat dipungkiri bahwa V.S.N Hardware Key Scheme dengan algoritma kriptografi RSA memiliki kelemahan yakni dengan serangan injeksi, namun hal ini dapat sedikit diatasi dengan V.S.N Illusion Scheme, dimana V.S.N yang ditanam di dalam storage drive bukan kunci otentifikasi langsung melainkan kunci hasil dari enkripsi menggunakan algoritma enkripsi yang mengedepankan aspek kemustahilan bukan kesulitan, yang dimana dalam proses otentifikasi membutuhkan pihak ketiga.

DAFTAR PUSTAKA

[1] Pfleeger, Charles, 2006, *Security in Computing*, New Jersey, Prentice Hall.
[2] Wang, Xiaoyun & Yu, Hongbo, 2005, *How to Break MD5 and Other Hash Functions*, Jinan.
[3] Nitaj, Abderrahmane, 2011, *A New Vulnerable Class of Exponents in RSA*, Caen.
[4] Wilson, Craig, 2005, *Volume Serial Numbers and Format Date/Time Verification*, Kent, Digital Detective Group.
[5] Rivest, R.L et al, 1977, *A Method for obtaining Digital Signatures and Public-Key Cryptosystems*, pp 1-15.
[6] Stallings, William, 2005, *Cryptography and Network Security Principles and Practices: 4th Edition*, Prentice Hall.
[7] Celko, Joe, 2001, *Data and Databases: Concepts in Practice*, San Francisco, Morgan Kauffman.