Jurnal Nasional Informatika dan Teknologi Jaringan
E-ISSN : 2540-7600
P-ISSN : 2540-7597

FRAMEWORK PENGAMANAN DATA DENGAN WHEEL FACTORIZATION PADA ALGORITMA RSA SEBAGAI PEMBANGKIT BILANGAN PRIMA

Oloan Sihombing
Magister Teknik Informatika, Universitas Sumatera Utara
Jl. Universitas No.9A Kampus USU, Medan, Sumatera Utara-Indonesia

***Abstrak*—Keamanan merupakan sebuah factor yang sangat penting di dalam pengiriman data. Banyak teknik keamanan data yang dapat digunakan untuk mengamankan data-data yang bersifat rahasia tersebut. Salah satunya adalah dengan menggunakan teknik kriptografi dengan menggunakan RSA. Akan tetapi di dalam metode tersebut kemungkinan metode tersebut dapat di retas tetap ada. Proses pembangkitan bilangan prima yang dibutuhkan di dalam metode RSA tersebut adalah proses yang paling utama sehingga proses peretasan akan semakin sulit. Di dalam penelitian ini akan memberikan sebuah framework baru di dalam teknik pengamanan data dengan RSA dengan menggunakan wheel factorization sebagai pembangkit bilangan primanya sehingga proses peretasan algoritma tersebut akan semakin sulit.**

***Keywords*— DOS, Denial Of Service, Keamanan Jaringan Komputer, firewall, Apache Server.**

I. PENDAHULUAN

Perkembangan komputer saat ini sangat pesat, dimana semua pekerjaan dilakukan secara sistem komputerisasi. Baik dalam pengiriman data atau pesan dari satu tempat ke tempat yang lain. Untuk itu perlu adanya dilakukan keamanan data dalam pengiriman informasi atau pesan, agar tidak dapat digunakan oleh orang-orang yang tidak bertanggungjawab. Masalah keamanan data merupakan salah satu aspek penting dari suatu sistem dengan teknologi informasi yang sedang berkembang pesat yang memungkinkan semua orang dapat berkomunikasi dari satu tempat ke tempat lain yang berjarak ribuan kilometer. Data yang dikirim itu menggunakan jalur transmisi telekomunikasi yang belum tentu terjamin keamanannya. Bisa saja data yang sedang dikirim melalui media transmisi itu dicuri atau diubah oleh penyadap dan hacker untuk kepentingan tertentu. Hal ini menjadi masalah bagi dunia telekomunikasi terutama dalam pengiriman data-data penting yang memerlukan kerahasiaan yang tinggi seperti informasi keuangan Bank, informasi rahasia negara dan informasi penting lainnya. Dalam menjaga kerahasiaan data, kriptografi mentransformasikan data jelas (plaintext) ke dalam bentuk data sandi (chipertext) yang tidak dapat dikenali oleh tanpa adanya password yang tepat. Chipertext inilah yang kemudian dikirim oleh pengirim (sender) kepada penerima (receiver). Setelah sampai di sipenerima, chipertext tersebut ditransformasikan kembali ke dalam bentuk plaintext agar dapat dipahami. Proses transformasi dari plaintext menjadi chipertext disebut proses Encipherment atau enkripsi (encryption), sedangkan proses mentransformasikan kembali chipertext menjadi plaintext disebut deskripsi (decryption).

Algoritma kriptografi modern tidak lagi mengandalkan keamanannya pada kerahasiaan algoritma tetapi kerahasiaan kunci. Plaintext yang sama bila disandikan dengan kunci yang berbeda akan menghasilkan chipertext yang berbeda pula. Dengan demikian algoritma kritografi dapat bersifat umum dan boleh diketahui oleh siapa saja, akan tetapi pengetahuan tentang kunci dan tersandi tetap saja tidak dapat dibaca atau dimanfaatkan.

Penyandian data itu dilakukan dengan pola-pola tertentu dengan menggunakan kunci rahasia yang hanya diketahui oleh pengirim dan penerima sehingga data rahasia itu bisa dideskripsi menjadi data asli

II. TINJAUAN PUSTAKA

A. Algoritma Dasar dan Sistem Kriptografi

Algoritma dasar kriptografi atau algoritma penyandian sebenarnya adalah suatu fungsi matematis yang digunakan untuk enkripsi dan dekripsi data. Untuk mengenkripsi dan mendekripsi data kriptografi menggunakan suatu algoritma (Chiper) dan Kunci (Key). Algoritma Kriptografi modern tidak lagi mengandalkan keamanannya pada kerahasiaan algoritma tetapi kerahasiaan kunci. Plaintext yang sama bila disandikan dengan kunci yang berbeda akan menghasilkan chipertext yang berbeda pula. Dengan demikian, algoritma kriptografi dapat bersifat umum dan boleh diketahui siapa saja, akan tetapi tanpa pengetahuan tentang kunci, data tersandi tetap saja tidak dapat dipecahkan. Sistem kriptografi adalah sebuah algoritma kriptografi ditambah semua kemungkinan plaintext, chipertext dan kunci [1][2][3].

B. Metode RSA (Rivest Shamir Adlemen)

Metode RSA merupakan salah satu bagian dari kriptografi modern, dimana metode RSA ini berdasarkan pada ide dengan mengalikan dua bilangan. Pada dasarnya proses dalam metode RSA ini melakukan pembangkit kunci untuk menghasilkan suatu bilangan prima sebagai nilai kunci dalam proses

penyandian data. Sebagai contoh, adalah relative mudah untuk mengambil dua bilangan prima p dan q dan menghitung hasil kalinya N = pq. Namun jika diberikan nilai N, sulit untuk menemukan faktornya p dan q, khususnya untuk bilangan N yang besar. Enkripsi menggunakan nilai atau kunci public (public key), yang disebarluaskan dan diketahui semua orang yang ingin mengirim pesan. Sedangkan dekripsinya menggunakan sebuah kunci pribadi (private key) yang dijaga kerahasiaanya oleh penerima dan tidak dapat dideduksi dari kunci publik. Kriptografi dengan kunci publik bekerja tanpa mengharuskan kedua pihak menjaga kerahasiaan. Kunci pribadi tidak pernah perlu diberitahukan ke pengirim pesan [4][5][6][7].

C. Algoritma Wheel Factorization

Algoritma Wheel Factorization merupakan metode grafis dari suatu bilangan bulat yang ditandai sekitar roda untuk membentuk jari-jari bilangan prima dan kelipatan bilangan. Bilangan prima adalah bilangan yang menarik untuk masyarakat matematika. Bilangan prima juga penting dalam algoritma enkripsi RSA sebagai pembangkit kunci. Proses dalam algoritma Wheel Factorization ini melakukan pemfaktoran suatu bilangan.

Wheel Factorization adalah metode untuk menyaring nomor komposit dengan mengatur nomor sekitar lingkaran dan kemudian mencolok keluar angka-angka yang jelas komposit seperti yang ditunjukkan oleh mana mereka jatuh di sekitar lingkaran [8][9][10].

D. Teknik Penyerangan Dalam RSA

Didalam teknik RSA dikenal beberapa teknik penyerangan yaitu [11][12][13]:

1. Known-Plain Text Analysisi, dengan prosedur ini, kriptanalis memiliki pengetahuan tentang sebagian dari plaintext dari ciphertext. Menggunakan informasi ini, kriptanalis berusaha untuk menyimpulkan kunci yang digunakan untuk menghasilkan ciphertext.
2. Choosen-Plaintext Analysis, juga dikenal sebagai kriptanalisis diferensial, cryptanalyst mampu memiliki plaintext dienkripsi dengan kunci dan mendapatkan ciphertext yang dihasilkan, tetapi kunci itu sendiri tidak dapat dianalisis. kriptanalis berusaha untuk menyimpulkan kunci dengan membandingkan seluruh ciphertext dengan plaintext asli. The Rivets-teknik enkripsi Shamir-Adleman telah terbukti menjadi agak rentan terhadap jenis analisis.
3. Cipher analisis teks, cryptanalyst tidak memiliki pengetahuan tentang plaintext dan harus bekerja hanya dari ciphertext. Ini membutuhkan menebak akurat bagaimana pesan dapat worded. Ini membantu untuk memiliki beberapa pengetahuan tentang gaya sastra dari penulis ciphertext dan / atau materi pelajaran umum.
4. Man-in-the-middle attack: Ini berbeda dari atas dalam hal itu melibatkan menipu individu ke dalam menyerahkan kunci mereka. The cryptanalyst / penyerang menempatkan dirinya dalam saluran komunikasi antara dua pihak yang ingin bertukar kunci mereka untuk komunikasi yang aman (melalui kunci asimetrik atau publickriptografi infrastruktur). Itucryptanalyst / penyerang kemudian melakukan pertukaran kunci dengan masing-masing pihak, dengan pihak asli percaya mereka bertukar kunci dengan satu sama lain. Kedua belah pihak kemudian berakhir dengan menggunakan kunci yang dikenal dengan cryptanalyst / penyerang. Jenis serangan bisa dikalahkan dengan menggunakan fungsi hash.
5. Timing/ Differential Power Analysis, ini adalah teknik yang baru diumumkan pada bulan Juni 1998, sangat berguna melawan kartu pintar, yang mengukur perbedaan dalam konsumsi listrik selama periode waktu ketika microchip melakukan fungsi untuk mengamankan informasi. Teknik ini dapat digunakan untuk mendapatkan informasi tentang perhitungan utama yang digunakan dalam algoritma enkripsi dan fungsi lainnya yang berkaitan dengan keamanan. Teknik ini dapat diberikan kurang efektif dengan memperkenalkan gangguan acak ke dalam perhitungan, atau mengubah urutan executable untuk membuat lebih sulit untuk memantau fluktuasi daya.

E. Penyerangan Dengan Faktorisasi

Penyerangan dengan teknik faktorisasi juga menjadi ancaman yang perlu di perhatikan, berikut merupakan beberapa contoh teknik faktorisasi

- Trail Division
- Pollard (P-1)
- Pollard Rho
- Fermat's Factoring method
- Quadratic Sieve Factoring

Trial Division adalah teknik tertua untuk faktorisasi di mana kita mulai membagi diberi nomor komposit dengan jumlah prima mulai dari 2,3,5,6 dan seterusnya. Jelas itu akan memakan waktu yang sangat besar untuk nomor komposit besar. Oleh karena itu tidak dapat digunakan untuk skenario dunia nyata.

algoritma faktorial berikutnya terkenal adalah algoritma Fermat Factorization. Ini adalah metode yang sangat sederhana dasarnya berdasarkan pada hubungan x2 - y2 = (x - y) (x + y).

Jika kita dapat menemukan y sehingga n + y2 = x2 maka (x - y) | n dan juga (x + y) | n. Metode Fermat bekerja dengan baik ketika faktor jumlah ini menjadi dua hal ukuran kira-kira sama yang faktor yang lebih dekat dengan akar kuadrat dari jumlah yang akan difaktorkan. Ia bekerja buruk ketika faktor-faktor yang sangat berbeda ukuran.

Pseudocode untuk Teorema Fermat diberikan sebagai

FermatFactor (N): // N harus ganjil
a ← ceil (sqrt (N))
b2 ← a * a - N sementara b2 tidak persegi:
a ← a + 1 // ekuivalen: b2 ← b2 + 2 * a + 1 b2 ← a
* a - N // a ← a + 1 end while return - sqrt (b2) // atau + sqrt (b2).

F. Faktorisasi

Dalam teori bilangan, faktorisasi integer adalah dekomposisi dari sejumlah komposit menjadi produk dari bilangan bulat kecil. Jika bilangan bulat ini lebih dibatasi untuk bilangan prima, proses ini disebut faktorisasi prima. Ketika jumlahnya sangat besar, tidak ada, non-kuantum algoritma faktorisasi integer efisien dikenal; upaya oleh beberapa peneliti menyimpulkan pada tahun 2009 [14][15][16][17].

Faltor sejumlah 232-digit (RSA-768), memanfaatkan ratusan mesin selama rentang dua tahun. Namun, belum terbukti bahwa tidak ada algoritma yang efisien ada. Kesulitan diduga dari masalah ini adalah jantung dari algoritma banyak digunakan dalam kriptografi seperti RSA. Banyak bidang matematika dan ilmu komputer telah dibawa untuk menanggung pada masalah, termasuk kurva berbentuk bulat panjang, nomor teori aljabar, dan komputasi kuantum. Tidak semua nomor dari panjang yang diberikan sama-sama sulit untuk faktor.

Contoh paling sulit dari masalah ini (untuk teknik saat ini dikenal) adalah semi bilangan prima, produk dari dua bilangan prima. Ketika mereka berdua besar, misalnya lebih dari dua ribu bit panjang, dipilih secara acak, dan tentang ukuran yang sama (tapi tidak terlalu dekat, misalnya, untuk menghindari faktorisasi efisien dengan metode faktorisasi Fermat), bahkan tercepat perdana faktorisasi algoritma pada tercepat komputer dapat mengambil cukup waktu untuk membuat pencarian lebih praktis, yaitu sebagai jumlah digit dari bilangan prima menjadi faktor meningkat, jumlah operasi yang dibutuhkan untuk melakukan faktorisasi pada setiap kenaikan computer secara drastis.

Banyak protokol kriptografi didasarkan pada kesulitan anjak bilangan bulat komposit besar atau masalah-untuk berhubungan misalnya, masalah RSA. Algoritma yang efisien faktor integer sewenang-wenang akan membuat RSAbased kriptografi aman kunci publik.

III. OBJECTIVES

Didalam penelitian ini Algoritma RSA merupakan algoritma kriptografi kunci publik (asimetris). Ditemukan pertama kali pada tahun 1977 oleh R. Rivest, A. Shamir, dan L. Adleman. nama RSA sendiri diambil dari ketiga penemunya tersebut. Sebagai algoritma kunci publik, RSA mempunyai dua kunci, yaitu kunci publik dan kunci rahasia. Kunci publik boleh diketahui oleh siapa saja, dan digunakan untuk proses enkripsi. Sedangkan kunci rahasia hanya pihak-pihak tertentu saja yang boleh mengetahuinya, dan digunakan untuk proses dekripsi. Keamanan sandi RSA terletak pada sulitnya memfaktorkan bilangan yang besar. Sampai saat ini RSA masih dipercaya dan digunakan secara luas di internet. Teknik pembangkit bilangan prima yang di gunakan di dalam teknik tersebut sudah digunakan cukup lama yang telah menjadi kadaluarsa dan menjadi sasaran bagi para peretas untuk meretas algoritma ini. Untuk mengatasi hal tersebut akan di lakukan modifikasi terhadap algoritma RSA ini dengan mengganti teknik pembangkit bilangan prima dengan mengguakan Wheel Factorization.

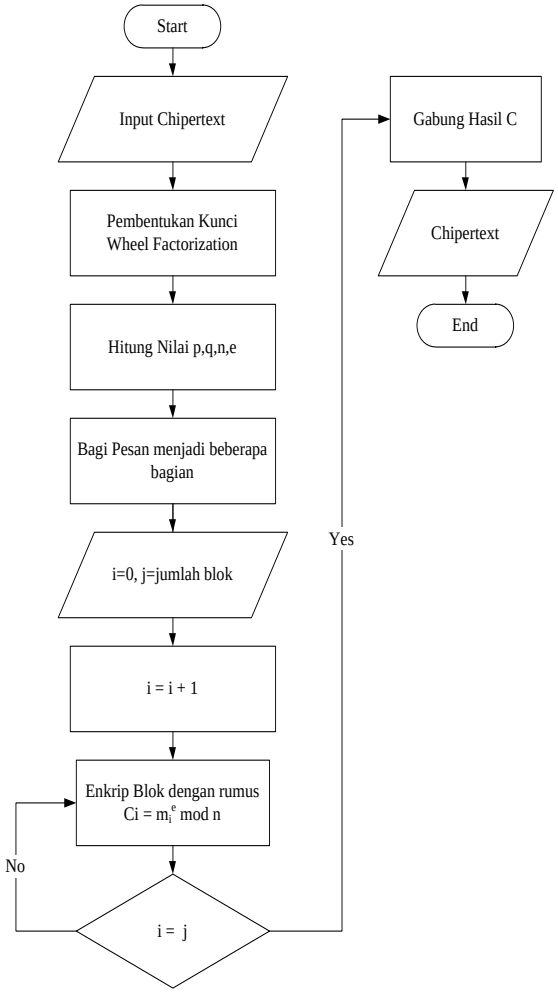
IV. METODE YANG DIUSULKAN

A. Enkripsi

Perancangan proses enkripsi pembentukkan kunci dengan menggunakan metode The Wheel Factorization. Penjelasan proses enkripsi pembentukkan kunci dengan menggunakan metode The Wheel Factorization dapat dilihat seperti berikut :berikut :

Langkah 1 : Input Plainteks
Langkah 2 : Pembentukkan kunci dengan metode The Wheel Factorization untuk mendapatkan bilangan prima.
Langkah 3 : Hitunglah nilai p dan q sebagai nilai n untuk mendapatkan kunci enkripsi.
Langkah 4 : Bagi pesan dalam file menjadi beberapa bagian.
Langkah 5 : Kelompokkan hasil pembagian pesan menjadi beberapa blok.
Langkah 6 : Hitunglah nilai
Langkah 7 : Blok-blok pesan yang telah di kelompokkan maka dienkrip sesuai rumus yang telah ditentukan.
Langkah 8 : Lakukan pengujian i = j
Jika ya, lakukan penggabungan hasil C
Jika tidak, lakukan langkah ke 7
Langkah 9 : Hasil ciphertext

Untuk Lebih jelasnya dapat dilihat pada gambar 1 untuk diagram alir dari proses enkripsi tersebut



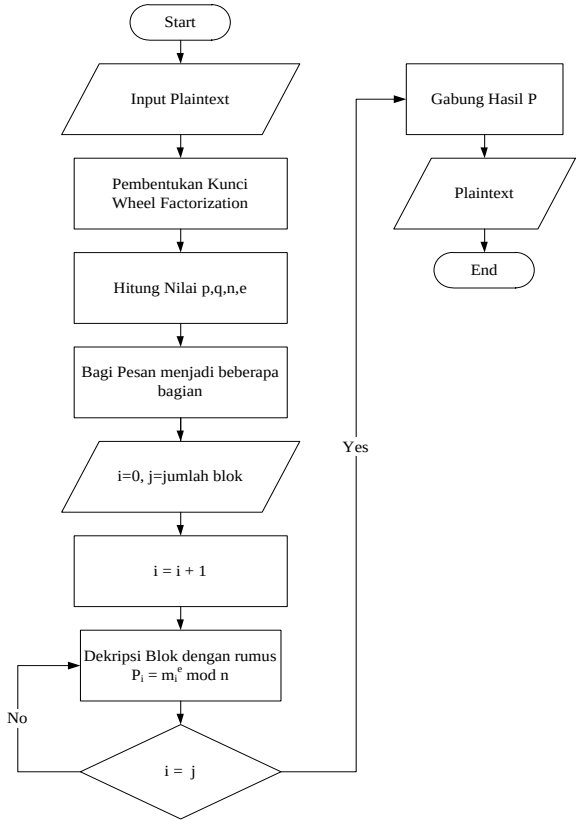
Gbr.1 Diagram Alir Proses Enkripsi

B. Denkripsi

Perancangan proses dekripsi pembentukkan kunci dengan menggunakan metode The Wheel Factorization. Penjelasan proses dekripsi pembentukkan kunci dengan menggunakan metode The Wheel Factorization dapat dilihat seperti berikut :

Langkah : Input Chipertext
Langkah 2 : Pembentukkan kunci dengan metode The Wheel Factorization untuk mendapatkan bilangan prima.
Langkah 3 : Hitunglah nilai p dan q sebagai nilai n untuk mendapatkan kunci dekripsi.
Langkah 4 : Bagi pesan dalam file menjadi beberapa bagian.
Langkah 5 : Kelompokkan hasil pembagian pesan menjadi beberapa blok.
Langkah 6 : Hitunglah nilai i
Langkah 7 : Blok-blok pesan yang telah di kelompokkan maka di dekripsi sesuai rumus yang telah ditentukan.
Langkah 8 : Lakukan pengujian i = j Jika ya, lakukan penggabungan hasil P Jika tidak, lakukan langkah ke 7
Langkah 9 : Hasil Plaintext

Untuk Lebih jelasnya dapat dilihat pada gambar 2 untuk diagram alir dari proses Dekripsi tersebut :



Gbr.2 Diagram Alir Proses Dekripsi

V. KESIMPULAN

Dari metode yang di usulkan tersebut terlihat bahwa proses pembangkitan bilangan prima yang di gunakan oleh RSA standar telah di gantikan dengan menggunakan teknik pembangkitan bilangan prima dengan menggunakan metode Wheel Factorization untuk pembangkit bilangan prima. Sehingga untuk mengamankan sebuah data dengan menggunakan teknik rsa tersebut akan menjadi lebih aman dengan menggunakan pembangkit kunci yang lebih rumit dan proses untuk meretas algoritma tersebut mungkin tetap ada akan tetapi akan lebih menghambat para peretas untuk memperoleh data rahasia tersebut.

REFERENSI

[1] W. Pan; F. Zheng; Y. Zhao; W. T. Zhu; J. Jing, "An Efficient Elliptic Curve Cryptography Signature Server with GPU Acceleration," in IEEE Transactions on Information Forensics and Security , vol.PP, no.99, pp.1-1

[2] X. Zhou, W. Gong, W. Fu and L. Jin, "An improved method for LSB based color image steganography combined with cryptography," 2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS), Okayama, Japan, 2016, pp. 1-4.

[3] S. S. Jathe and V. Dhamdhere, "Hybrid Cryptography for Malicious Behavior Detection and Prevention System for MANETs," 2015 International Conference on Computational Intelligence and Communication Networks (CICN), Jabalpur, India, 2015, pp. 1108-1114.

[4] S. Goyal, M. Ramaiya and D. Dubey, "Improved Detection of 1-2-4 LSB Steganography and RSA Cryptography in Color and Grayscale Images," 2015 International Conference on

Computational Intelligence and Communication Networks (CICN), Jabalpur, India, 2015, pp. 1120-1124.
[5] D. Aggarwal; U. Maurer, "Breaking RSA Generically is Equivalent to Factoring," in IEEE Transactions on Information Theory , vol.PP, no.99, pp.1-1
[6] R. Verma, M. Dutta and R. Vig, "FPGA implementation of RSA based on carry save Montgomery modular multiplication," 2016 International Conference on Computational Techniques in Information and Communication Technologies (ICCTICT), New Delhi, 2016, pp. 107-112.
[7] M. Garg, S. Gupta and P. Khatri, "Fingerprint watermarking and steganography for ATM transaction using LSB-RSA and 3-DWT algorithm," 2015 International Conference on Communication Networks (ICCN), Gwalior, 2015, pp. 246-251.
[8] William Stallings, "Cryptography and Network Security - Principles and Practice", Fifth Edition, Prentice Hall, ISBN: 978-0-13-609704-4. [2]. Hans Riesel, "Prime Numbers and Computer methods for factorization", Progress in Mathematics, Vol.57, ISBN: 0-8176-3291-3.
[9] Santanu Sarkar, "Some Results on Cryptanalysis of RSA and Factorization", PhD thesis, ISI Kolkata, 2011
[10] K. Mori, T. Nguyen, T. Harada and R. Thawonmas, "An Improvement of Matrix Factorization with Bound Constraints for Recommender Systems," 2016 5th IIAI International Congress on Advanced Applied Informatics (IIAI-AAI), Kumamoto, Japan, 2016, pp. 103-106.
[11] S. J. Aboud, "An efficient method for attack RSA scheme," Applications of Digital Information and Web Technologies, 2009. ICADIWT '09. Second International Conference on the, London, 2009, pp. 587-591.
[12] N. N. Albassam and M. Nasereddin., "Solution Space Optimization for RSA Attack," Developments in eSystems Engineering (DeSE), 2013 Sixth International Conference on, Abu Dhabi, 2013, pp. 243-246.
[13] F. Jia and D. Xie, "A unified method based on SPA and timing attacks on the improved RSA," in China Communications, vol. 13, no. 4, pp. 89-96, April 2016.
[14] C. L. Duta, L. Gheorghe and N. Tapus, "Framework for evaluation and comparison of integer factorization algorithms," 2016 SAI Computing Conference (SAI), London, United Kingdom, 2016, pp. 1047-1053.
[15] D. F. G. Coelho, R. J. Cintra, S. Kulasekera, A. Madanayake and V. S. Dimitrov, "Error-free computation of 8-point discrete cosine transform based on the Loeffler factorisation and algebraic integers," in IET Signal Processing, vol. 10, no. 6, pp. 633-640, 8 2016.
[16] H. Yu and G. Bai, "An efficient method for integer factorization," 2015 IEEE International Symposium on Circuits and Systems (ISCAS), Lisbon, 2015, pp. 73-76.
[17] S. Sarnaik, R. Bhakkad and C. Desai, "Comparative study on Integer Factorization algorithm-Pollard's RHO and Pollard's P-1," Computing for Sustainable Global Development (INDIACom), 2015 2nd International Conference on, New Delhi, 2015, pp. 677-679.pan