Analisis Pola Identifikasi Zero Knowledge Proof Dengan Algoritma Feige Fiat Shamir Menggunakan Blum Blum Shub

[1)] Cherlina Helena Purnamasari Panjaitan
STIKOM Medan; Jalan Jamin Ginting No 285 Medan, Sumatera Utara Indonesia
E-Mail: helenpanjaitan01@gmail.com

[2)] Lisda Juliana Pangaribuan
AMIK MBP Medan; Jalan Jamin Ginting No 285 Medan, Sumatera Utara, Indonesia
E-Mail: lisdajuliana@gmail.com

ABSTRACT
Protocol Zero Knowledge Proof is one of the protocols in Cryptography that has a fairly good level of security, because it applies the concept of "Truly Zero Knowledge Proof" which is not leaking any information. This protocol is used in the Fiat Shamir, Guillou Quisquater and Schnorr Feige Algorithms, all of which are Cryptographic Algorithms using private keys and public keys. In the Public key, all three of these Algorithms use a random number generator at the values p and q to get the public key. In this study, the author will generate a public key generation test using CPRNG (Cryptographically-secure Pseudo-Random Number Generator) with the Blum Blum Shub algorithm. The test will be conducted on the Fiat Feige Algorithm, the formation of the key will use the Blum Blum Shub Algorithm, but the Identification Protocol still uses the Fiat Shamir Feige Algorithm. The results of this study show the Feige Fiat Algorithm with the Blum Blum Shub Algorithm as the key builder successfully identifies the pattern sent by the signer.

Keyword : Zero Knowledge Proof, Feige Fiat Shamir, CPRNG, Blum Blum Shub, Cryptography

PENDAHULUAN

Berbagai algoritma telah dikembangkan untuk melindungi informasi, salah satunya melalui metode pengamanan informasi menggunakan password. Password berfungsi untuk mengamankan informasi yang dilindungi, sehingga hanya user yang memiliki password yang tepat yang berhak mendapatkan dan mengelola informasi tersebut.[1] Bentuk lain dari metode password adalah dengan kode pin, yang banyak ditemukan pada ponsel, kartu ATM, dan perangkat elektronik lainnya. Selain itu, terdapat pengamanan menggunakan tanda tangan digital, sehingga data yang diamankan dapat teruji keasliannya, serta tidak dapat diubah oleh pihak yang tidak berkepentingan.[2] Algoritma untuk keamanan informasi dikembangkan untuk melindungi informasi tersebut agar tidak dirusak,diubah atau dicuri oleh pihak yang tidak berkewenangan.

Telah banyak Algoritma yang ditemukan dan dikembangkan oleh para ahli, khususnya algoritma yang berhubungan dengan keamanan komputer, mulai dari algoritma yang sederhana sampai yang bersifat kompleks. Namun yang terpenting dalam sebuah sistem keamanan bukan hanya algoritmanya saja, melainkan juga protokol yang mengaturnya. Di dalam kriptografi, terdapat berbagai jenis protokol. Disini, penulis akan membahas mengenai salah satu jenis protokol kriptografi, yaitu Zero Knowledge Proof Protokol dan algoritma pola identifikasinya. Ada tiga algoritma untuk identifikasi dengan Protokol Zero Knowledge Proof, salah satunya yaitu Feige Fiat Shamir.[3]

Protokol Pola Identifikasi Zero Knowledge Proof merupakan solusi untuk mengurangi kebocoran password dari pihak ketiga yang bermaksud mencuri hak mengakses informasi yang tersimpan. Dilihat dari sistematikanya, Protokol Zero Knowledge Proof memberikan jaminan bahwa pihak verifier dapat membuktikan identitas pihak prover tanpa mempelajari informasi rahasia sedikitpun dari yang diketahui oleh prover. Protokol Zero Knowledge Proof membuat pengguna tidak perlu kuatir bahwa pihak lain, dalam hal ini bisa juga adalah server, menyalahgunakan kunci yang dimilikinya.[4] Protocol Zero Knowledge Proof merupakan sebuah Protokol Kriptografi dimana seseorang (prover) dapat membuktikan kepemilikannya atas suatu informasi rahasia atau identitasnya kepada pihak ketiga (verifier) dengan tanpa membocorkan seluruh informasi rahasia tersebut tanpa memberikan cara bagi orang lain untuk mengetahui rahasia tersebut. Protokol ini dikembangkan oleh Uriel Feige, Amos Fiat, dan Andi Shamir dengan konsep “Truly Zero Knowledge Proof” yaitu tidak membocorkan informasi apapun.[1]

Algoritma yang mengusung Protokol ini adalah Pola Identifikasi Feige Fiat Shamir. Uriel Fiege, Amos Fiat dan Adi Shamir memodifikasi skema otentikasi dan digital signature yang

MEANS (Media Informasi Analisa dan Sistem)
p-ISSN : 2548-6985, e-ISSN : 2599-3089

dikembangkan oleh Amos Fiat dan Adi Shamir menjadi sebuah identitas pembuktian zero-knowledge. Skema Feige Fiat Shamir merupakan salah satu skema penandaan digital dengan menerapkan konsep Zero Knowledge Proof.[4] Algoritma ini menggunakan kunci asimetris atau kunci publik.[5] Dalam makalah mereka, Feige, Fiat dan Shamir menunjukkan bagaimana konstruksi paralel dapat menambah jumlah dari akreditas per putaran dan mengurangi interaksi antara kedua orang yang sedang berkomunikasi.[6] Algoritma ini menggunakan teori square roots untuk meningkatkan keamanan dari algoritma mereka. Amannya algoritma Feige Fiat Shamir dilihat dari tingkat kesukaran pada proses kalkulasi akar kuadrat modulo pq, bilamana p dan q tidak teridentifikasi (n = pq saja yang teridentifikasi). Uriel Fiege, Amos Fiat dan Adi Shamir memodifikasi skema otentikasi dan digital signature yang dikembangkan oleh Amos Fiat dan Adi Shamir menjadi sebuah identitas pembuktian zero-knowledge. Algoritma ini merupakan identitas pembuktian zero-knowledge yang terbaik saat ini.[7]

Di dalam dunia Kriptografi terdapat berbagai modifikasi, Salah satu modifikasinya adalah dengan menambahkan proses matematis pada saat men-generate bilangan acak yang akan digunakan sebagai kunci proses enkripsi dan dekripsi. Blum Blum Shub merupakan salah suatu metode yang berfungsi men-generate bilangan acak secara matematis, dengan menghasilkan output deretan angka biner. Modifikasi kedua Algoritma ini yang menjadi objek penelitian dalam jurnal ini.

Blum Blum Shub juga salah satu generator pada CPRNG yang sangat sederhana dan terbukti aman, kemudian dalam hubungannya dengan kualitas generator, dianggap kuat karena sulitnya faktorisasi integer. Blum Blum Shub memiliki bentuk persamaan: [8]

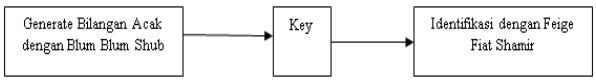
$$X_{n+1} = X_{n2} \bmod m \quad \ldots\ldots\ldots\ldots\ldots\ldots (1)$$

dimana m merupakan hasil dari perkalian dua buah bilangan prima besar p dan q, serta output-nya dalam Least Significant Bit dari Xn dimana hal yang sama sebagai parity dari Xn. Dua buah bilangan prima p dan q harus kongruen terhadap 3 mod 4 dan Greatest Common Divisor (GCD) harus kecil. Pada aplikasi Kriptografi, Generator ini sering digunakan karena tidak begitu cepat.

METODE PENELITIAN

Tahapan yang dilakukan pada penelitian ini adalah Pengumpulan data, analisis data, perancangan sistem dan Implementasi sistem.
Analisis akan mengacu pada keempat tujuan mendasar dari kriptografi yang juga merupakan aspek keamanan informasi, yaitu: Kerahasiaan (confidentiality/privacy), [9](integrity) dan Identifikasi (identification) dan Otentikasi (authentication). Nonrepudiation.[9]
Langkah-langkah analisis masalah adalah Mengenerate Random Number Blum-Blum Shub, Mengenerate Key kemudian proses identifikasi dari skema identifikasi Feige-Fiat-Shamir.

Langkah-langkah analisis masalah dalam penelitian dapat dilihat seperti **Gambar** 1.



Gambar 1. Langkah-langkah analisis masalah

A. Generate Random Number Blum-Blum Shub

Proses kerja dari skema Feige Fiat Shamir akan dilakukan menjadi dua bagian, yaitu Tahapan pembentukan kuncI dan tahapan kerja protokol identifikasi. Proses pembentukan kunci dari skema identifikasi Feige-Fiat-Shamir ini adalah[7] :

1. Tentukan 2 nilai bilangan prima besar, yaitu p , q.
2. Lakukan perhitungan nilai n = p * q.
3. Ambil k buah bilangan V dimana V berupa quadratic residue modulo n, yaitu x2 ≡ V (mod n) harus memiliki hasil nilai x berupa bilangan bulat dan V-1 (mod n) harus memiliki hasil. Perhitungan V-1 (mod n) akan memiliki hasil jika GCD(Vi, n) = 1.
4. Hitung nilai si terkecil dimana V-1 (mod n) ≡ si2 (mod n).
5. Public key : v1, v2, …, vk
6. Private key : s1, s2, …, sk

Proses pembentukan kunci ini dilakukan oleh Signer.

Proses identifikasi dari skema identifikasi Feige-Fiat-Shamir ini adalah[3] :

1. Signer memilih sebuah nilai bilangan yang acak, r, dimana r lebih kecil dibandingkan n. Kemudian, dia melakukan perhitungan dengan rumus x = r2 mod n dan mengirim x pada Verifier.
2. Verifier mengirim deretan bit biner acak dengan panjang k-bit, yaitu b1, b2, …, bk.
3. Signer menghitung y = r * (s1b1 * s2b2 * … * skbk) mod n. Dia mengirimkan y kepada Verifier.
4. Verifier memverifikasi bahwa x = y2 * (V1b1 * V2b2 * … * Vkbk).
5. Jika sama, maka valid. Jika tidak sama, maka tidak valid.
6. Ulangi proses diatas sebanyak t kali sampai Verifier merasa yakin bahwa Signer memang mengetahui s1, s2, …, sk.

B. Generate Key

1) Bilangan Prima

Bilangan prima banyak digunakan untuk pembangkitan Kunci publik dalam kriptografi. Sebuah bilangan dapat dikatakan sebuah bilangan prima jika merupakan suatu bilangan integer yang memiliki nilai lebih tinggi daripada 1, kemudian mempunyai faktor bilangan satu serta bilangan tersebut sendiri. Pemakaian bilangan prima adalah hal penting pada sebuah

MEANS (Media Informasi Analisa dan Sistem)
p-ISSN : 2548-6985, e-ISSN : 2599-3089

matematika kriptografi, digunakan untuk pembuatan kunci publik.[10]

Terdapat cara yang tidak tepat untuk mengambil sebuah bilangan prima yaitu melakukan pembangkitan bilangan acak lalu melakukan pemfaktoran. Selanjutnya cara yang lebih tepat untuk mendapatkan bilangan prima adalah melalui melakukan pembangkitan bilangan acak dan setalah itu melakukan tindakan pengetesan apakah bilangan tersebut adalah bilangan prima atau tidak.

2) Pembangkit Bilangan Acak (Cryptographically-secure Pseudo-Random Number Generator)

Dapat didefinisikan sebagai suatu peralatan komputasional, perancangannya adalah untuk memberikan hasil suatu urutan nilai yang memiliki pola tidak sukar untuk ditebak, urutan nilai yang dimaksud adalah keadaan acak (random), ini adalah definisi dari Cryptographically-secure Pseudo-Random Number Generator (CPRNG). Sebenarnya, bilangan acak hasil dari komputer tidaklah sungguh-sungguh teracak, demikian juga bilangan random kebanyakan yang terdapat di dalam kriptografi pun tidak sungguh-sungguh random, namun hanyalah semu. Dengan kata lain kemungkinan bilangan acak yang ada tersebut dapat diketahui susunan serta urutannya[11].

Pada kriptografi dilakukan pembangkitan bilangan acak dengan pembangkit bilangan acak semu ini. Kemudian (CPRNG) akan mengeluarkan hasil urutan nilai yang elemen-elemennya dipengaruhi pada setiap nilai yang dihasilkan. Seperti telah dibahas sebelumnya, hasil dari CPRNG sebenarnya tidak sesungguhnya acak, Namun ada kemiripan dengan properti dari nilai acak. Semakin rumit CPRNG nya maka tingkat keamanan dari metoda kriptografi semakin baik[6].

3) Blum Blum Shub (BBS)

Langkah-langkah pada algoritma Blum Blum Shub adalah : [8]

1. Lakukan pemilihan 2 bilangan prima yang dirahasiakan, p,q, masing-masing harus kongruen dengan 3 modulo 4.
2. Lakukan perkalian kedua bilangan menjadi n = pq. Bilangan n ini dikatakan bilangan bulat Blum
3. Lakukan pemilihan bilangan bulat acak lainnya, s, sebagai umpan, sehingga:
   2 £ s < n dimana dan n relatif prima
   Selanjutnya lakukan perhitungan x0 = s2 mod n.
4. Baris bit acak akan dihasilkan dengan dilakukannya iterasi berikut sepanjang yang diinginkan: Lakukan perhitungan xi = xi – 1 2 mod n
   zi = bit LSB (Least Significant Bit) dari xi
   Hasil baris bit acak adalah z1, z2, z3, …

Tingkat keamanan pada metode Blum Blum Shub adalah karena sulitnya memfaktorkan n yang dihasilkan BBS. Tidak perlu merahasiakan nilai n, artinya nilai n dapat dipublikasikan kepada publik. Barisan bit acak pada metode BBS tidak dapat diprediksi dari arah kiri (unpredictable to the left) dan arah kanan (unpredictable to the right), sehingga dapat disimpulkan, apabila terdapat barisan bit yang dihasilkan dengan BBS, kriptanalis tidak dapat memprediksi barisan bit sebelumnya dan barisan bit sesudahnya.

C. Proses identifikasi dari skema identifikasi Feige-Fiat-Shamir

Penelitian akan dilanjutkan dengan memodifikasi proses pembangkitan kunci pada Algoritma Feige Fiat Shamir. Proses pembangkitan kunci pada algoritma Feige Fiat Shamir menggunakan bilangan acak prima, penulis akan memodifikasi dengan menggunakan Metode Blum Blum Shub pada proses pembangkitan kunci privat dan kunci publiknya. Penulis akan membandingkan proses pembangkitan kunci algoritma Feige Fiat Shamir dengan proses pembangkitan kunci algoritma Feige Fiat Shamir menggunakan metode Blum Blum Shub sebagai pembangkit bilangan acaknya.[7]

HASIL DAN PEMBAHASAN

a.Pengujian

Pengujian Menggunakan Algoritma Blum Blum Shub.

Proses Pembentukan Kunci:

1. Tentukan dua buah bilangan prima yang kongruen 3 modulo 4, yaitu p dan q
   Nilai p = 11 dan Nilai q = 23
2. Hitung nilai n= 253;
3. Nilai k = 4;
4. Nilai V(1) = 169, V(2) = 64, V(3) = 163, V(4) = 225.
5. Hitung nilai invers VI(1) = 3, VI(2) = 70,VI(3)= 104, VI(4) = 9.
6. Nilai S(1) = 16; S(2) = 26; S(3) = 37;S(4) = 3.
   Kunci Publik : V(1) = 169; V(2) = 64; V(3) = 163; V(4) = 225
   Kunci Privat :
   S(1) = 16, S(2) = 26, S(3) = 37, S(4) = 3

Proses Identifikasi:

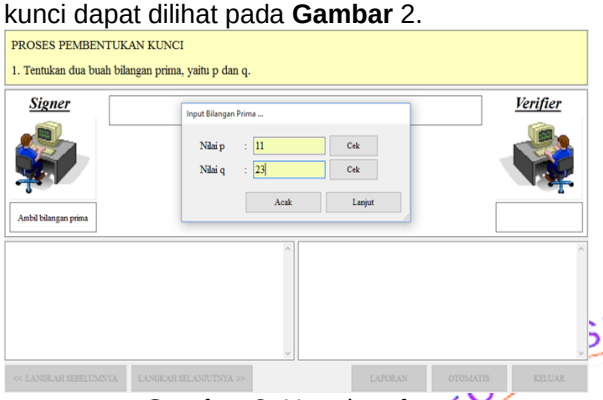
1. Nilai r = 127
2. Nilai x = 190
3. Random bit string dengan panjang k bit, mengacu pada Algoritma Blum Blum Shub, nilai bit biner String dihasilkan dari perhitungan :
   Memilih p = 11 dan q = 23 sehingga n = pq = 253.
   Pilih s = 3 dan hitung x0 = 32 mod 253 = 9.
   Barisan bit acak dihasilkan dengan melakukan iterasi berikut sepanjang k:
   x1 = x02 mod n = 92 mod 253 = 81 è z1 = 1 (karena 81 ganjil, bit LSB-nya pasti 1)

x2 = x12 mod n = 812 mod 253 = 236 è z2 = 0 (karena 236 genap, bit LSB-nya pasti 0)
x3 = x12 mod n = 2362 mod 253 = 36 è z2 = 0 (karena 36 genap, bit LSB-nya pasti 0)
x4 = x12 mod n = 362 mod 253 = 31 è z2 = 0 (karena 31 ganjil, bit LSB-nya pasti 1)
Barisan bit acak yang dihasilkan 1001
b = 1001

4. Nilai y = 24
5. Nilai Xt = 190
   X [190] = Xt [190] --> Proses Identifikasi SUKSES !!!

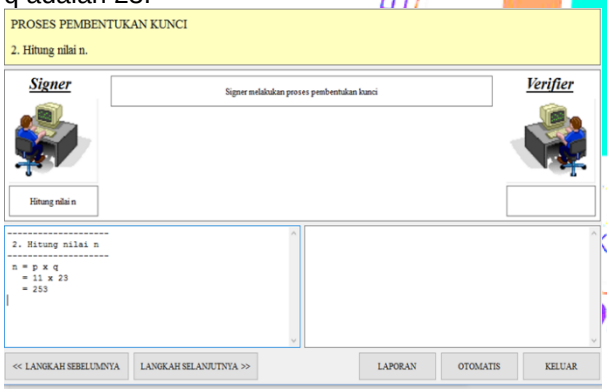
b.Implementasi Sistem

Pada system yang dibangun proses pembentukan kunci dapat dilihat pada **Gambar** 2.



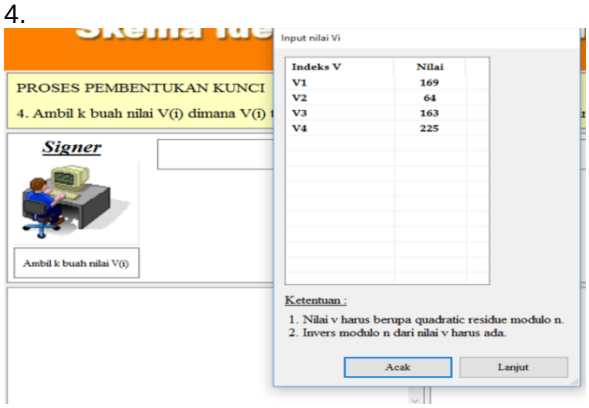
Gambar 2. User interface

Dari **Gambar** 2 dapat dilihat bahwa Signer menginput bilangan prima 11 untuk nilai p dan nilai q adalah 23.

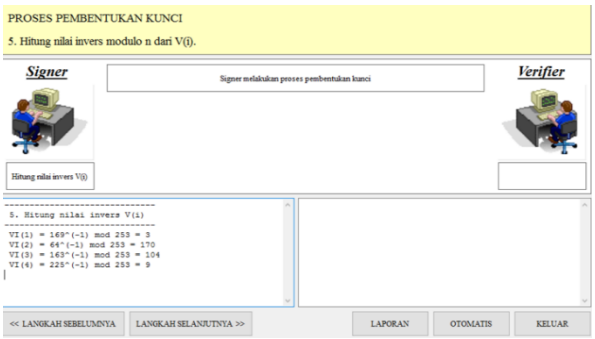


Gambar 3. Proses Pembentukan kunci

Pada **Gambar** 3 dari hasil input p dan q diperoleh nilai n = 253 dan kemumudian ditentukan nilai k = 4.

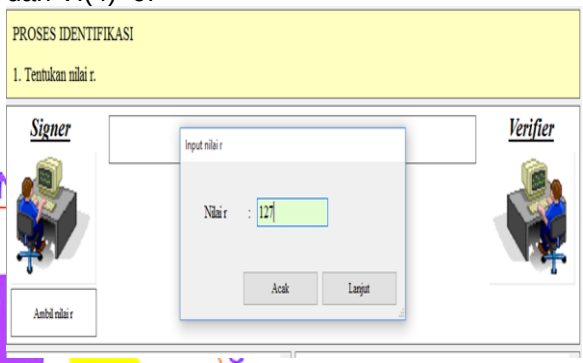


Gambar 4. Input Nilai V(i)



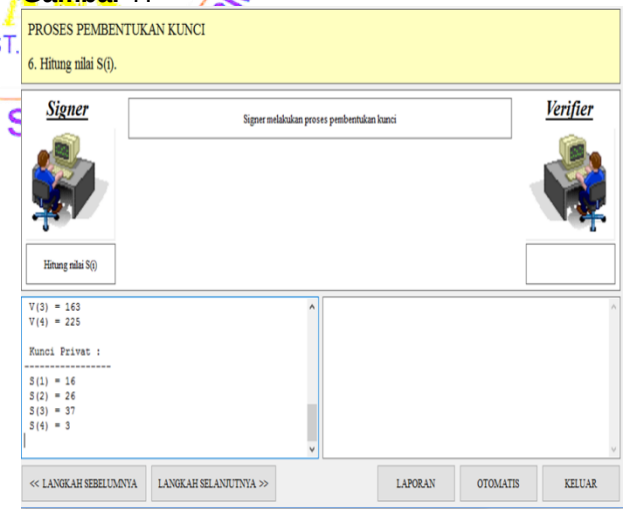
Gambar 5. Nilai Invers VI

Dari **Gambar** 4. dapat dilihat nilai V yang diinput masing-masing adalah V1=169, V2=64, V3=163 dan V4=225.
Selanjutnya nilai invers VI) dapat dilihat pada **Gambar** 5. yaitu VI(1)=3, VI(2)=170, VI(3)=194 dan VI(4)=9.



Gambar 6. Kunci Privat

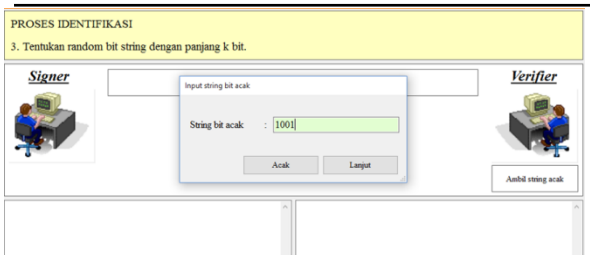
Gambar 6 menunjukkan kunci privat yang dipentuk dari hasil perhitungan adalah S(1) = 16, S(2)=26, S(3)= 37 dan terakhir S(4)=3.
Setelah memperoleh kunci privat, maka dilakukan proses identifikasi yang dapat dilihat pada **Gambar** 7.



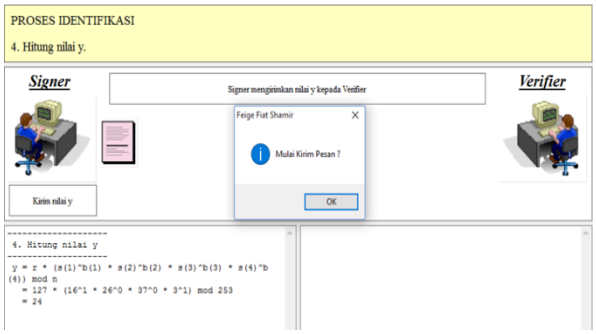
Gambar 7. Proses Identifikasi

Pada proses indentifikasi Signer mulai mengirim pesan yang akan diindentifikasi yang dimulai dengan memasukkan Nilai r=127 seperti pada **Gambar** 7 kemudian sistem otomatis menghitung nilai x= 190.

MEANS (Media Informasi Analisa dan Sistem)
p-ISSN : 2548-6985, e-ISSN : 2599-3089

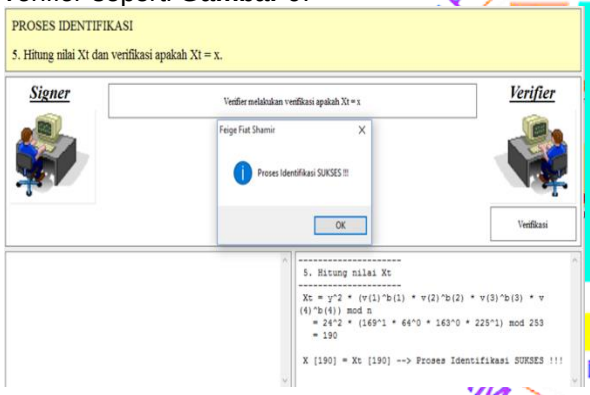


Gambar 8. Random Bit String



Gambar 9 Signer mengirim Nilai Feight Fist Shamir

Setelah nilai x diperoleh, *verifier* menginput String bilangan acak= 1001 seperti **Gambar** 8, lalu mengirimnya kepada *signer* kemudian diperoleh nilai y= 24 yang akan dikirim kembali kepada *verifier* seperti **Gambar** 9.



Gambar 10. Proses Verifikasi

Dari **Gambar** 10 dapat dilihat bahwa setelah dilakukan verifikasi nilai Xt=x yaitu 190. Dengan demikian proses Identifikasi Sukses artinya Pola dapat diindetifikasi.

KESIMPULAN

Setelah melakukan pengujian dan analisa dapat disimpulkan bahwa dengan pembangkitan kunci Algoritma Blum Blum Shub dan melakukan proses identifikasi dengan Algoritma Feige Fiat Shamir, maka ditemukan hasil yang lebih aman karena pada pembangkitan kunci dengan Algoritma Blum Blum Shub, menggunakan bilangan Prima yang harus kongruen 3 modulo 4 sedangkan Pembangkit kunci Feige Fiat Shamir hanya cukup menggunakan bilangan prima saja. Sebagai catatan, tidak semua bilangan prima kongruen 3 modulo 4.

DAFTAR PUSTAKA

[1] W. Sudiarto Raharjo, I. D. E.K. Ratri, and H. Susilo, “Implementasi Two Factor Authentication Dan Protokol Zero Knowledge Proof Pada Sistem Login,” *J. Tek. Inform. dan Sist. Inf.*, vol. 3, no. 1, pp. 127–136, 2017, doi: 10.28932/jutisi.v3i1.579.

[2] Y. Anshori, A. Y. Erwin Dodu, and D. M. P. Wedananta, “Implementasi Algoritma Kriptografi Rivest Shamir Adleman (RSA) pada Tanda Tangan Digital,” *Techno.Com*, vol. 18, no. 2, pp. 110–121, 2019, doi: 10.33633/tc.v18i2.2166.

[3] W. S. Raharjo and D. Sutanti, “Implementasi Zero Knowledge Proof Menggunakan Protokol Feige Fiat Shamir Untuk Verifikasi Tiket Rahasia,” *J. Ultim.*, vol. 7, no. 2, pp. 91–97, 2016, doi: 10.31937/ti.v7i2.355.

[4] R. Toyib and Y. Darnita, “Pengamanan Data Teks Dengan Menggunakan Algoritma Zero-Knowledge Proof,” no. 1, pp. 16–23, 2020.

[5] L. J. Pangaribuan, “Kriptografi Hybrida Algoritma Hill Cipher Dan Rivest Shamir Adleman (RSA) Sebagai Pengembangan Kriptografi Kunci Simetris (Studi Kasus : Nilai Mahasiswa Amik Mbp),” *J. Teknol. Inf. DAN Komun.*, vol. 7, no. 1, pp. 11–26, 2018.

[6] M. B. Sanjaya and P. A. Telnoni, “Implementasi Random Number Blum-Blum-Shub Dan Chaotic Function Untuk Modifikasi Key Generating Pada Kriptografi Aes Implementasi Blum-Blum-Shub Dan Chaotic Function Untuk Modifikasi Key Generating Pada Aes Implementation of Blum-Blum-Shub and Chaotic Func,” *J. Elektro Telekomun. Terap. Desember*, vol. 1, no. 1, pp. 154–165, 2015.

[7] L. Maulana, A. Kusyanti, and F. A. Bakhtiar, “Implementasi Metode Autentikasi dengan Zero Knowledge Proofmenggunakan Protokol Feige-Fiat-Shamir Identification Schemepada PerangkatInternet of Things,” vol. 3, no. 9, pp. 8937–8945, 2019, [Online]. Available: http://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/6311/3034.

[8] C. Harahap, G. Ginting, and T. Zebua, “Perancangan Aplikasi Pengacakan Pemenang Undian Berhadiah Menggunakan Metode Blum-Blum Shub Berbasis Android,” *JURIKOM (Jurnal Ris. Komputer)*, vol. 7, no. 2, p. 301, 2020, doi: 10.30865/jurikom.v7i2.2112.

[9] L. J. Pangaribuan, “Implementasi Kriptografi Menggunakan Metode Hill Chiper,” in *Seminar Nasional Inovasi dan Teknologi Informasi (SNITI)*, vol. 2014, pp.

1–9.

[10] L. J. Pangaribuan, "KRIPTOGRAFI MODERN KUNCI ASIMETRIS DENGAN METODE RSA UNTUK KEAMANAN PESAN DALAM E-MAIL," in *Konferensi Nasional Pengembangan Teknologi Informasi dan Komunikasi (KETIK)*, 2014, pp. 153–159.

[11] A. Ramadhan and S. Teknik, "Perbandingan Algoritma Linear Congruential Generators , BlumBlumShub , dan MersenneTwister untuk Membangkitkan Bilangan Acak Semu."