EDUMATSAINS
JURNAL PENDIDIKAN MATEMATIKA DAN SAINS
E-ISSN:2527-7235
Volume 8 Issue. 2, Janury, 2024, pp 196-216
http://ejournal.uki.ac.id/index.php/edumatsains

NPWP ENCRYPTION USES A COMBINATION OF *ONE TIME PAD* ALGORITHM AND *EXCLUSIVE-OR OPERATION*

Margaretha Pasaribu[1*], Sherli Yurinanda[2], Niken Rarasati[3]

[1,2,3]Matematika FST Universitas Jambi
e-mail: *** sherliyurinanda@unja.ac.id**,

Article Info
Article history:
Received: July 20, 2023
Revised: January 30, 2024
Accepted: January 30, 2024
Available online: January 31, 2024

https://doi.org/10.33541/edumatsains.v8i1.4576

Abstract
Nomor Pokok Wajib Pajak (NPWP) is the 20 distinct code characteristics that are confidential only to the taxpayer when it comes to making tax transactions. The many tax payers who use NPWP others in their business activities make NPWP a very important aspect of maintaining confidentiality and the authenticity of their data. Therefore, solutions are needed that can be used to make the necessary information secure. Cryptography involves the secret science of code writing using certain mathematical models. The study discusses NPWP encryption using a combination algorithm One Time Pad (OTP) and an operation Exclusive-OR (XOR). The encryption process involves generating random numbers for an OTP algorithm, after which it is followed by an XOR operation. The key on XOR encryption is generated from an algorithm generating random numbers Blum Blum Shub. Encryption keys generated as much as the number of plaquic characters. The encryption keys used by each character would vary and not occur again. Therefore, ciphertext results from an encrypted process using an OTP algorithm and an XOR operation more complicated and more difficult to decipher by crypts.

Keywords: NPWP, Cryptography, One Time Pad, Exclusive-OR, Blum Blum Shub

1. Introduction

Berdasarkan definisi yang dikemukakan oleh Halim et al (2007) Salah satu tugas Wajib Pajak adalah melakukan pendaftaran untuk mendapatkan Nomor Pokok Wajib Pajak (NPWP). Nomor Pokok Wajib Pajak adalah nomor yang diterima wajib pajak guna menjalankan urusan administrasi perpajakan untuk identifikasi diri, dimana Wajib Pajak hanya diberikan satu Nomor Wajib Pajak saja. Berdasarkan Pasal 28 Ayat 1 Ayat 6 UU KUP 2007 Nomor Pokok Wajib Pajak merupakan 15 digit kode unik. Kode unik ini berfungsfi menjamin informasi wajib pajak tidak tercampur dengan wajib pajak lainnya. Berdasarkan pengamatan Kanwil DJP Kaltimtara dalam mediasuara.com yang diakses pada tanggal 10 Maret 2023 terhitung banyak wajib pajak yang memakai NPWP atau faktur pajak orang lain Tidak Berdasarkan kepada Transaksi Sebenarnya

(TBTS) dalam usaha yang dilakukan. Hal ini cenderung menimbulkan kerugian bagi kedua wajib pajak, baik pemilik NPWP maupun pemakainya. Sehingga perlu dijaga keamanan datanya. Melihat bahaya penyalahgunaan data pribadi maka penulis menggunakan kriptografi sebagai layanan keamanan data. Secara umum, algoritma enkripsi belum sempurna, namun untuk mendapatkan algoritma yang lebih aman dan kemungkinan solusi yang lebih sedikit, ada One Time Pad (OTP). OTP berisi serangkaian karakter kunci yang dibuat secara acak. Jumlah karakter dalam kunci sesuai dengan jumlah karakter dalam pesan. Satu pad digunakan hanya satu kali (one time) guna mengamankan pesan (Ariyus, 2008). Guna memperkuat keamanan data penulis mengkombinasikan algoritma OTP dengan operasi Exclusive-OR (XOR) dalam mangamankan Nomor Pokok Wajib Pajak (NPWP). Menurut Sidik et al (2019) dimana karakter bit pada plainteks di-XOR dengan setiap karakter bit kunci. Teknik XOR menawarkan keunggulan kecepatan yang besar karena teknik XOR merupakan algoritma dengan efisiensi waktu yang baik dalam pemrosesan enkripsi dan dekripsi lebih cepat dibandingkan dengan algoritma enkripsi lainnya. Menurut hasil penelitian yang dilakukan oleh Kumar et al (2012) Keamanan teknik XOR dapat dinaikkan dengan cara menambah jumlah kunci dan mengacak kunci secara acak. Cara memperbanyak kunci yang dimiliki dalam teknik XOR berdasarkan kajian yang telah dilakukan oleh Sanjaya (2017) mengatakan bahwa guna menaikan kerandoman kunci yang dipakai, diperlukan algoritma khusus untuk menghasilkan angka acak. Algoritma yang digunakan penulis adalah algoritma Blum-BlumShub.

2. Methods

Penelitian ini merupakan studi literatur, jenis penelitian adalah penelitian kuantitatif dengan menggunankan information sekunder. Sumber information pada penelitian ini diterbitkan oleh Direktorat Jenderal Pajak berupa Nomor Pokok Wajib Pajak Orang Pribadi. Information yang dikumpulkan dalam penelitian ini bersifat deskriptif. Dikutip dari laman resmi npwponline.com Nomor Pokok Wajib Pajak (NPWP) berisi 15 digit, yakni 9 digit awal adalah kode Wajib Pajak dan 6 digit selanjutnya adalah kode untuk administrasi perpajakan dengan format XX.XXX.XXX.X-XXX.XXX.

1. Keterbagian

Keterbagian adalah bagian dasar dari karakteristik teori bilangan. Misalkan dua bilangan c, d $\in \mathbb{Z}$ dengan c $\neq 0$, maka c disebut membagi d dinotasikan c|d apabila d = ck, untuk suatu $k \in \mathbb{Z}$ (Irawan, 2014).

2. Aritmatika Modulo

Aritmatika modulo merupakan operasi aritmatika yang memetakan semua bilangan bulat ke himpunan bilangan bulat di dalam batas-batas himpunan. Misalkan $a \in \mathbb{Z}$ dan $n \in \mathbb{Z}$ dan $n > 0$.
Notasi : $a\ (mod\ n) \equiv r$ sedemikian sehingga $a = nq + r$, dengan $0 \leq r < n$. (Stallings, 2003).

3. Pembagi Bersama Terbesar (PBB)

Pembagi bilangan bulat positif terbesar dapat membagi kedua bilangan bulat a dan b, yang disebut pembagi bersama terbesar (PBB) (Batten, 2012).

Bilangan bulat b dan c (b dan c bukan nol) maka bilangan bulat a dikatakan pembagi bersama terbesar dari b dan c jika memenuhi (Irawan, 2014):

1. $a > 0$
2. $a|b$ dan $a|c$
3. Jika $d|b$ dan $d|c$ maka $d|a$

Notasi : Pembagi bersama terbesar dari b dan c dituliskan $a = (b, c)$ dan karena $a > 0$ maka $a = (b, c) \geq 1$ sehingga $(b, c) = (b, -c) = (-b, c) = (-b, -c)$.

4. Kongruensi

Kongruensi adalah ide pembagian bilangan bulat pada teori bilangan dengan memakai konsep kongruensi yang ada. Misalkan $m \in \mathbb{N}$ dan $a, b \in \mathbb{Z}$ dikatakan kongruen jika m membagi $(a - b)$ dapat ditulis : $a \equiv b \ (mod\ m)$. Pada kasus ini, m dikatakan modulo kongruensi dan a kongruen dengan b (Das, 2013).

5. Kriptografi

kriptografi merupakan ilmu dan seni guna melindungi keamanan pesan. Pengirim ingin pesan tetap aman dalam pengiriman, dimana pihak lain tidak dapat membaca atau memodifikasi pesan teks terenkripsi. Pesan yang dikodekan sedemikian rupa menjadi tidak memiliki makna. Tujuannya agar orang yang tidak berwenang tidak dapat membaca pesan tersebut. Enkripsi merupakan proses menyandikan plainteks menjadi cipherteks. Sedangkan dekripsi merupakan proses membalikan cipherteks menjadi plainteks awal.

Notasi Matematis (Munir, 2006)

$$E(P) = C$$

$$D(C) = P$$

Fungsi enkripsi dan dekripsi memenuhi sifat:

$$D(E(P)) = P$$

Keterangan:

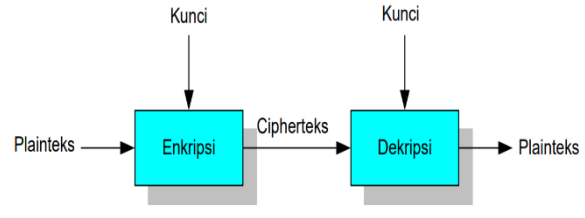
C = cipherteks

P = plainteks

$E(P) =$ fungsi enkripsi yang memetakan elemen elemen dari plainteks

$D(C) =$ fungsi dekripsi yang memetakan elemen-elemen dari cipherteks

Menurut Munir (2006) kunci merupakan ukuran yang digunakan untuk mengubah enkripsi dan dekripsi. Kunci bersifat rahasia (mystery), sedangkan algoritma kriptografi tidak rahasia (open). Kriptografi dibagi ke dalam algoritma kriptografi klasik dan algoritma kriptografi modern. Algoritma kriptografi klasik secara umum menggunakan kunci-simetri. Kunci-simetri adalah kunci identic yang digunakan pada saat proses enkripsi dan dekripsi, berikut cara kerjanya pada **Gambar 1**:



Gambar 1. Proses Enkripsi Dan Dekripsi
(Sumber: Munir, 2006)

a. One Time Pad

Menurut Munir (2006) One Time Pad termasuk algoritma unbreakable cipher. Unbreakable cipher adalah klaim yang dibuat oleh kriptografer terhadap algoritma kriptografi yang dibuatnya. Syarat untuk enkripsi yang tidak dapat dipecahkan (unbreakable cipher) adalah kuncinya sangat acak dan panjang kunci sama dengan panjang plainteks.

Akibatnya, plainteks yang sama tidak pasti terenkripsi menjadi ciphertext yang sama. Pada hal ini Penerima teksnharuslah memiliki salinan pad yang sama, Dimana sebuah pad hanya digunakan sekali (*one time*) untuk mengenkripsi pesan.

$$Panjang\ kunci\ OTP\ =\ Panjang\ plainteks$$

Sehingga tidak ada pengulangan penggunaan kunci selama proses pengamanan pesan. Aturan enkripsi yang dipakai:

$$Enkripsi: c_i = (p_i + k_i)\ mod\ n \quad ...(1)$$
$$Dekripsi: p_i = (c_i - k_i)\ mod\ n \quad ...(2)$$

Keterangan:

$c_i =$ karakter cipherteks ke $- i$
$p_i =$ karakter plainteks ke $- i$
$k_i =$ karakter kunci ke $- i$
$n =$ jumlah karakter yang digunakan

b. Exclusive-OR

Operasi *XOR* merupakan operasi logika bitwise yang berjalan dengan cara membandingkan dua buah bit (0 dan 1) yang disajikan dalam **Tabel 1**:

Tabel 1. XOR atau operasi $\oplus$

Input x	y	**Output** (x, y)
0	0	0
0	1	1
1	0	1
1	1	0

Menurut (Ariyus, 2008) Operasi biner merupakan proses menghubungkan atau memetakan sebuah himpunan ke himpunan itu sendiri menggunakan operasi biner. Bilangan Biner ini merupakan bilangan yang terdiri dari 2 karakter, yakni angka 0 dan angka 1.

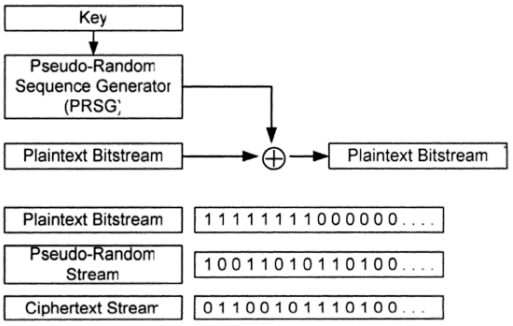
c. Aliran kode

Operasi penjumlahan modulo 2 linear dengan operasi bit dengan Operator XOR maka persamaan (3) dapat ditulis menjadi:

$$c_i = p_i \oplus k_i \quad \text{... (5)}$$

Dan proses dekripsi menggunakan persamaan:

$$p_i = c_i \oplus k_i \quad \text{... (6)}$$



Gambar 2. Proses Aliran Kode

(Sumber: Ariyus, 2008)

d. Pembangkit Bilangan Acak

Angka acak adalah karakter penting dalam kriptografi dan dapat disebut sebagai angka yang kemunculan dan nilainya tidak dapat diperkirakan. Ada dua jenis angka acak: Yang pertama adalah angka acak sejati, yaitu bilangan acak yang tidak dapat ditelusuri kembali. angka acak kedua adalah angka semi-acak (pseudo-random number). Jenis bilangan acak ini dihasilkan dengan metode perhitungan (algoritma) berdasarkan beberapa parameter yang disebut sebagai seed, yang bertindak sebagai kunci. Generator angka semi-acak disebut pseudorandom number generator (PRNG). Generator angka semi-acak memiliki fungsi periodik, yang berarti bahwa urutan angka acak yang dihasilkan berulang dari awal setelah satu periode atau lebih. Oleh karena sifat ini, bilangan semi acak bersifat deterministik (Munir, 2019).

a. Blum-Blum Shub Generator

Blum-Blum Shub (BBS) adalah generator bilangan acak semu yang diusulkan tahun 1986 oleh Lenore Blum, Manuel Blum, dan Michael Shub. (Sidik, 2019).

Menurut Sanjaya (2017) Algoritma generator bilangan acak BBS sebagai berikut:

1. Pilih dua buah bilangan prima rahasia p dan q, dimana masing-masing kongruen 3 $modulo$ 4 atau $3 \equiv x \, mod \, 4$
2. Kalikan keduanya sehingga $n = pq$. Dimana n disebut bilangan bulat Blum.
3. Pilih bilangan acak lain s, untuk umpan sedemikian sehingga:

   (i) $2 \leq s \leq n$

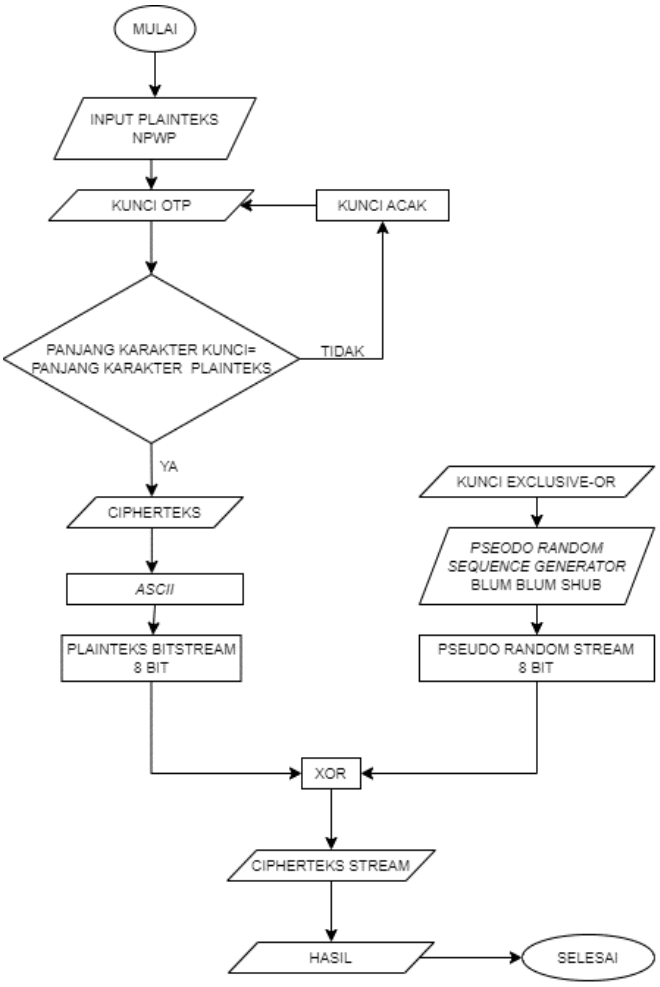
   (ii) s dan n relatif prima

   Kemudian hitung $x_0 = s^2 \, mod \, n$
4. Barisan bit acak dihasilkan dengan menjalankan iterasi sebanyak yang diperlukan.

   (iii) hitung $x_i = {x_{i-1}}^2 \, mod \, n$, x_i adalah bilangan acak ke-i

   (iv) z_i= gabungan *Least Significant Bit (LSB)* dari x_i

   Barisan bit acak yang dihasilkan menjadi $z_1, z_2, z_3, ..$



Gambar 3. Diagram Alir Enkripsi Algoritma

3. Result and Discussion

1. Proses Enkripsi

Pada penelitian ini penulis ingin memberikan gambaran tentang cara kerja algoritma *One Time Pad (OTP)* dan operasi *Exclusive-OR (XOR)* dalam menyandikan pesan asli berupa Nomor Pokok Wajib Pajak (NPWP) orang pribadi yakni 65.524.711.2-331.000 yang akan diubah menjadi pesan rahasia (enkripsi). Pesan asli atau plainteks disandikan dengan menggunakan algoritma *OTP* dilanjutkan dengan operasi *XOR*. Sebelum proses enkripsi dan dekripsi, angka acak dibangkitkan dan digunakan sebagai kunci.

1.1 Enkripsi *One Time Pad*

Pembangkitan bilangan acak sebagai kunci pada algoritma *OTP* harus memenuhi syarat, Panjang kunci angka acak harus sesuai dengan panjang teks terenkripsi atau pesan asli. Pada penelitian ini kunci yang dibangkitkan secara acak untuk proses enkripsi dekripsi *OTP* adalah 37428593154962376139. Terdapat sebuah plainteks yang akan diubah menjadi cipherteks berupa Nomor Pokok Wajib Pajak yang terdiri dari 20 karakter yaitu: 65.524.711.2-331.000, yang diamankan dengan algoritma *One Time Pad*. Karakter plainteks dan kunci yang akan dienkripsi yaitu:

Plainteks: 65.524.711.2-331.000

Kunci: 37428593154962376139

Langkah yang harus dilakukan untuk mengenkripsi NPWP serta menjalankan algoritma OTP adalah mengkonversi karakter plainteks dan kunci menjadi bentuk *American Standard Code for Information Interchange (ASCII)*. Dalam hal ini karakter plainteks ke-1 yaitu angka 6 dikonversi dalam *ASCII* dengan indeks karakter 54 lalu karakter kunci ke-1 yaitu angka 3 dikonversi dalam ASCII dengan indeks karakter 51. Indeks karakter inilah yang digunakan dalam perhitungan proses enkripsi menggunakan algoritma *OTP* dengan banyak karakter sesuai jumlah indeks karakter *ASCII* yaitu 256 karakter yang ditulis sebagai berikut:

$$(54 + 51)\ mod\ 256 = 105$$

Hasil enkripsi NPWP menggunakan algoritma *OTP* pada digit ke-2 sampai ke-20 disajikan pada **Tabel 2.**

Tabel 2. Enkripsi *One Time Pad*

P	***ASCII***	***K***	***ASCII***	***ASCII***	***C***
6	54	3	51	105	i
5	53	7	55	108	l
.	46	4	52	98	b
5	53	2	50	103	g
2	50	8	56	106	j

4	52	5	53	105	i
.	46	9	57	103	g
7	55	3	51	106	j
1	49	1	49	98	b
1	49	5	53	102	f
.	46	4	52	98	b
2	50	9	57	107	k
-	45	6	54	99	c
3	51	2	50	101	e
3	51	3	51	102	f
1	49	7	55	104	h
.	46	6	54	100	d
0	48	1	49	97	a
0	48	3	51	99	c
0	48	9	57	105	i

Keterangan:

P = plainteks

K = kunci

C = cipherteks

1.2 Enkripsi *Exclusive-OR*

Agar plainteks semakin sulit dipecahkan maka proses enkripsi dilanjutkan dengan operasi *Exclusive-OR (XOR)*. Plainteks yang digunakan dalam *XOR* merupakan hasil enkripsi NPWP menggunakan algoritma *One Time Pad (OTP)*. Hasil enkripsi *OTP* masih dalam bentuk indeks karakter *ASCII* sehingga perlu dikonversi menjadi bilangan biner 8 bit agar dapat dienkripsi menggunakan operasi *XOR.* Sebagai contoh pada digit ke-1 hasil enkripsi *OTP* dengan indeks karakter *ASCII* yaitu 105 dikonversi dengan cara sebagai berikut:

$$105 : 2 = 52\ sisa\ 1$$

$$52 : 2 = 26\ sisa\ 0$$

$$26 : 2 = 13\ sisa\ 0$$

$$13 : 2 = 6\ sisa\ 1$$

$$6 : 2 = 3\ sisa\ 0$$
$$3 : 2 = 1\ sisa\ 1$$
$$1 : 2 = 0\ sisa\ 1$$

Dengan demikian didapatkan bentuk biner dari 105 adalah 01101001. Proses konversi pada karakter ke-2 sampai ke-20. Sesudah itu dilakukan proses pembangkitan kunci untuk enkripsi *XOR* menggunakan generator angka acak Blum Blum Shub (BBS) dengan langkah langkah sebagai berikut:

1. Pilihlah dua buah bilangan prima rahasia, yaitu p dan q yang mana masing-masing kongruen dengan 3 mod 4 berikut ini :

   $p \equiv 3\ mod\ 4$ maka $p\ mod\ 4 = 3$, sehingga diambil $p = 79$

   $q \equiv 3\ mod\ 4$ maka $q\ mod\ 4 = 3$, sehingga diambil $q = 83$

2. Mengalikan p dan q untuk mencari nilai n

   $n = p \times q$

   $n = 79 \times 83$

   $n = 6557$

3. Memilih bilangan bulat lain menjadi nilai umpan yaitu s $(seed)$ dengan syarat s dan n relative prima dimana s dan n memiliki faktor prima 1

   $2 \leq 29 < n$

   $n = 6557$ dengan $FPB = 1,59,73$

   $s = 29$ dengan $FPB = 1$

   Setelah diketahui bahwa n dan s relatif prima, selanjutnya hitung x_0 :

   $x_0 = s^2\ mod\ n = 29^2\ mod\ 6557 = 841\ mod\ 6557 = 841$

4. Mencari *Least Signifacant Bit (LSB*) untuk menghasilkan karakter kunci, ketentuan untuk menghasilkan nilai *LSB* adalah jika hasil $mod\ x_i$ bernilai ganjil, maka nilai *LSB* yang dihasilkan adalah 1. Jika nilai $mod\ x_i$ bernilai genap, maka nilai *LSB* yang dihasilkan adalah 0. Proses untuk mendapatkan hasil nilai $mod\ x_2$ berdasarkan pada hasil x_1 atau hasil x_i sebelumnya. Demikian juga untuk mendapatkan nilai $mod\ x_i$ selanjutnya selalu merujuk pada

nilai $mod\ x_{i-1}$ atau nilai sebelumnya.

Proses pembentukan karakter kunci sebagai berikut:

$x_1 = {x_0}^2\ mod\ n = 29^2\ mod\ 6557 = 841\ mod\ 6557 = 841; LSB = 1$

$x_2 = {x_1}^2\ mod\ n = 841^2\ mod\ 6557 = 707281\ mod\ 6557 = 5682; LSB = 0$

$x_3 = {x_2}^2\ mod\ n = 5682^2\ mod\ 6557 = 32285124\ mod\ 6557 = 5013; LSB = 1$

$x_4 = {x_3}^2\ mod\ n = 5013^2\ mod\ 6557 = 25130169\ mod\ 6557 = 3745; LSB = 1$

$x_5 = {x_4}^2\ mod\ n = 3745^2\ mod\ 6557 = 14025025\ mod\ 6557 = 6159; LSB = 1$

$x_6 = {x_5}^2\ mod\ n = 6159^2\ mod\ 6557 = 37939381\ mod\ 6557 = 1036; LSB = 0$

$x_7 = {x_6}^2\ mod\ n = 1036^2\ mod\ 6557 = 1073296\ mod\ 6557 = 4505; LSB = 1$

$x_8 = {x_7}^2\ mod\ n = 4505^2\ mod\ 6557 = 20295025\ mod\ 6557 = 1110; LSB = 0$

Dengan demikian didapat 1 karakter kunci BBS yaitu 10111010. Bilangan biner 8-bit diperlukan untuk membentuk 1 karakter. Karena NPWP terdiri dari 20 karakter, pencarian dilakukan untuk nilai x_i hingga 160 (8×20). Berikut ini adalah kunci yang dihasilkan dengan membangkitkan angka acak menggunakan generator Blum Blum Shub yang disajikan dalam bentuk *binary* pada **Tabel 3.**

Tabel 3. Bilangan Acak $\boldsymbol{X_I}$

Kelompok Biner Kunci	Hasil Gabungan *LSB*
$1(x_1 - x_8)$	10111010
$2(x_9 - x_{16})$	11001110
$3(x_{17} - x_{24})$	10000100
$4(x_{25} - x_{32})$	10011111
$5(x_{33} - x_{40})$	01001001
$6(x_{41} - x_{48})$	10100001
$7(x_{49} - x_{56})$	01101011
$8(x_{57} - x_{64})$	00101011
$9(x_{65} - x_{72})$	10101100
$10(x_{73} - x_{80})$	11101000
$11(x_{81} - x_{88})$	01001001
$12(x_{89} - x_{96})$	11110100
$13(x_{97} - x_{104})$	10011010
$14(x_{105} - x_{112})$	00010110

$15(x_{113} - x_{120})$	10110010
$16(x_{121} - x_{128})$	10111010
$17(x_{129} - x_{136})$	11001110
$18(x_{137} - x_{144})$	10000100
$19(x_{145} - x_{152})$	10011111
$20(x_{153} - x_{160})$	01001001

Dengan demikian didapatkan 20 karakter kunci operasi *XOR*. Dengan menggunakan Operasi *XOR* maka diperoleh output berupa bilangan biner dalam **Tabel 4.**

Tabel 4. Operasi *XOR*

Input		Output
Plainteks	**kunci**	
0	1	1
1	0	1
1	1	0
0	1	1
1	1	0
0	0	0
0	1	1
1	0	1

Dengan demikian didapat cipherteks 11010011.

Hasil enkripsi disajikan pada **Tabel 5.**

Tabel 5. Enkripsi Menggunakan Operasi *XOR*

p	***xor***	**k**	***C***
01101001	⊕	10111010	11010011
01101100	⊕	11001110	10100010
01100010	⊕	10000100	11100110
01100111	⊕	10011111	11111000
01101010	⊕	01001001	00100011
01101001	⊕	10100001	11001000
01100111	⊕	01101011	00001100
01101010	⊕	00101011	01000001
01100010	⊕	10101100	11001110
01100110	⊕	11101000	10001110
01100010	⊕	01001001	00101011
01101011	⊕	11110100	10011111
01100011	⊕	10011010	11111001

01100101	⊕	00010110	01110011
01100110	⊕	10110010	11010100
01101000	⊕	10111010	11010010
01100100	⊕	11001110	10101010
01100001	⊕	10000100	11100101
01100011	⊕	10011111	11111100
01101001	⊕	01001001	00100000

Tabel 6. Konversi Biner Ke Bentuk Karakter

Biner	Indeks karakter	karakter
11010011	211	Ë
10100010	162	ó
11100110	230	µ
11111000	248	°
00100011	35	#
11001000	200	╚
00001100	12	FF
01000001	65	A
11001110	206	╬
10001110	142	Ä
00101011	43	+
10011111	159	ƒ
11111001	249	¨
01110011	115	S
11010100	212	I
11010010	210	Ê
10101010	170	¬
11100101	229	Õ
11111100	252	³
00100000	32	SP

Dengan demikian dari proses enkripsi Nomor Pokok Wajib Pajak 65.524.711.2-331.000 menggunakan algoritma *One Time Pad* dan Operasi *Exclusive-OR* maka NPWP diamankan menjadi data yang tidak bisa dimengerti lagi yaitu Ëóµ°#╚$FF\ A╬Ä + ƒ¨SIÊ¬Õ³SP.$

2. **Proses Dekripsi**

Nomor Pokok Wajib Pajak (NPWP) telah disandikan menjadi karakter yang tidak bermakna lagi, hal ini dilakukan agar NPWP tidak terbaca oleh pihak yang tidak memiliki hak. Pihak yang

bisa membacanya haruslah mempunyai kunci, dimana kunci yang dijalankan dalam penelitian ini adalah kunci simetri sehingga untuk menjalankan proses dekripsi digunakan kunci yang sama dengan saat proses enkripsi sebelumnya. Dekripsi ini adalah proses mengembalikan cipherteks Ëóµ°#╚***FF A***╬Ä + ƒ***SI***Ê¬Õ³***SP*** menjadi bentuk semula.

2.1 Dekripsi *Exclusive-OR*

Dalam melakukan operasi *XOR* ada beberapa hal yang akan dilakukan agar pesan dapat didekripsi. Pada penelitian ini peneliti menjalankan proses dekripsi pada hasil enkripsi NPWP menggunakan algoritma *One Time Pad* dan operasi *exclusive OR*. Pesan rahasia *(ciphertext*) akan dikembalikan menjadi pesan semula (*plaintext)*. Pesan rahasia yang digunakan adalah Ëóµ°#╚*FF A*╬Ä + ƒ*SI*Ê¬Õ³*SP*.

Dalam melakukan operasi *XOR* dilakukan proses konversi untuk Menentukan indeks karakter dari cipherteks. Untuk karakter cipherteks ke-1 yaitu Ë perlu dikonversi ke dalam bentuk indeks karakter *ASCII***,** dimana untuk karrakter Ë memiliki indeks karakter 211, untuk karakter cipherteks ke-2 yaitu ó memiliki indeks karakter 162 seterusnya hingga didapat indeks karakter cipherteks ke-20 yang disajikan pada **Tabel 7.**

Tabel 7. Konversi Plainteks Ke Bentuk Indeks Karakter

Cipherteks	***ASCII***
Ë	211
ú	162
µ	230
°	248
#	35
╚	200
FF	12
A	65
╬	206
Ä	142
+	43
ƒ	159
¨	249
s	115
ı	212

Ê	210
¬	170
Õ	229
³	252
SP	32

Selanjutnya menentukan biner dari indeks karakter. Berdasarkan **Tabel 7** karakter cipherteks ke-1 yaitu Ë memiliki indeks karakter 211, indeks inilah yang akan dikonversi ke dalam bentuk biner dengan cara sebagai berikut:

Indeks karakter: 211

$$211 : 2 = 105, \text{ sisa } 1$$

$$105 : 2 = 52, \text{ sisa } 1$$

$$52 : 2 = 26, \text{ sisa } 0$$

$$26 : 2 = 13, \text{ sisa } 0$$

$$13 : 2 = 6, \text{ sisa } 1$$

$$6 : 2 = 3, \text{ sisa } 0$$

$$3 : 2 = 1, \text{ sisa } 1$$

$$1 : 2 = 0, \text{ sisa } 1$$

Dengan demikian nilai biner dari 211 adalah 11010011. Hasil konversi disajikan pada **Tabel 8.**

Tabel 8. Konversi Indeks Karakter Ke Bentuk Biner

Indeks Karakter	Biner
211	11010011
162	10100010
230	11100110
248	11111000
35	00100011
200	11001000
12	00001100
65	01000001
206	11001110
142	10001110
43	00101011
159	10011111

249	11111001
115	01110011
212	11010100
210	11010010
170	10101010
229	11100101
252	11111100
32	00100000

Melakukan operasi *XOR*

Operasi *XOR* untuk karakter ke-1 sebagai berikut:

Cipherteks: 11010011

Kunci: 10111010

Perhitungan modulo:

$$(1+1) mod\ 2 = 0$$
$$(1+0) mod\ 2 = 1$$
$$(0+1) mod\ 2 = 1$$
$$(1+1) mod\ 2 = 0$$
$$(0+1) mod\ 2 = 1$$
$$(0+0) mod\ 2 = 0$$
$$(1+1) mod\ 2 = 0$$
$$(1+0) mod\ 2 = 1$$

Dengan demikian didapat plainteks 01101001.

Selanjutnya dilakukan enkripsi Operasi *XOR* pada **Tabel 9.**

Tabel 9. Operasi *XOR*

Input		**Output**
$\boldsymbol{p}$	$\boldsymbol{k}$	$(\boldsymbol{p}, \boldsymbol{k})$
1	1	0
1	0	1
0	1	1
1	1	0
0	1	1
0	0	0

1	1	0
1	0	1

Dengan demikian didapat plainteks 11010011. Hasilnya dapat dilihat pada **Tabel 10.**

Tabel 10. Dekripsi Menggunakan Operasi *XOR*

c	⊕	***K***	***P***
11010011	⊕	10111010	01101001
10100010	⊕	11001110	01101100
11100110	⊕	10000100	01100010
11111000	⊕	10011111	01100111
00100011	⊕	01001001	01101010
11001000	⊕	10100001	01101001
00001100	⊕	01101011	01100111
01000001	⊕	00101011	01101010
11001110	⊕	10101100	01100010
10001110	⊕	11101000	01100110
00101011	⊕	01001001	01100010
10011111	⊕	11110100	01101011
11111001	⊕	10011010	01100011
01110011	⊕	00010110	01100101
11010100	⊕	10110010	01100110
11010010	⊕	10111010	01101000
10101010	⊕	11001110	01100100
11100101	⊕	10000100	01100001
11111100	⊕	10011111	01100011
00100000	⊕	01001001	01101001

Setelah didapatkan hasil dekripsi operasi *XOR* berupa bilangan biner 8 bit maka ditentukan indeks karakternya. Untuk karakter plainteks hasil operasi *XOR* ke-1 yaitu 01101001 dikonversi menjadi indeks karakter, dimana indeks karakternya adalah 105 dengan karakter I. Hasil konversi disajikan pada **Tabel 11.**

Tabel 11. Konversi Biner Ke Bentuk Indeks Karakter

C	***ASCII***	***P***
01101001	105	*i*
01101100	108	*l*
01100010	98	*b*
01100111	103	*g*
01101010	106	*j*
01101001	105	*i*

01100111	103	*g*
01101010	106	*j*
01100010	98	*b*
01100110	102	*f*
01100010	98	*b*
01101011	107	*k*
01100011	99	*c*
01100101	101	*e*
01100110	102	*f*
01101000	104	*h*
01100100	100	*d*
01100001	97	*a*
01100011	99	*c*
01101001	105	*i*

Berdasarkan **Tabel 11** telah didapatkan indeks karakter dari proses dekripsi menggunakan operasi *XOR*. Selanjutnya dilakukan proses dekripsi menggunakan algoritma *OTP*.

2.2 Dekripsi *One Time Pad*

Proses dekripsi cipherteks menggunakan algoritma *OTP* dengan kunci acak yang sudah diketahui yakni:

Dekripsi *One Time Pad*

Kunci: 37428593154962376139

Selanjutnya untuk proses dekripsi cipherteks langkah yang harus dilakukan adalah mengkonversi karakter cipherteks menjadi bentuk *American Standard Code for Information Interchange (ASCII)*. Dalam hal ini karakter kunci ke-1 yaitu angka 3 dikonversi dalam *ASCII* dengan indeks karakter 51. Indeks karakter inilah yang digunakan dalam proses dekripsi menggunakan algoritma *OTP* dengan banyak karakter sesuai jumlah indeks karakter *ASCII* yaitu 256 karakter yang ditulis sebagai berikut:

$$(105 - 51) mod\ 256 = 54$$

Untuk proses enkripsi NPWP pada digit ke-2 sampai ke-20 disajikan pada **Tabel 12.**

Tabel 12. Proses Dekripsi Menggunakan Algoritma *OTP*

Proses Dekripsi	**Proses Dekripsi**

$(105-51) mod\ 256 = 54$	$(98-52)\ mod\ 256 = 46$
$(108-55)\ mod\ 256 = 53$	$(107-57)\ mod\ 256 = 50$
$(98-52)\ mod\ 256 = 46$	$(99-54)\ mod\ 256 = 45$
$(103-50)\ mod\ 256 = 53$	$(101-50)\ mod\ 256 = 51$
$(106-56)\ mod\ 256 = 50$	$(102-51)\ mod\ 256 = 51$
$(105-53) mod\ 256 = 52$	$(104-55)\ mod\ 256 = 49$
$(103-57) mod\ 256 = 46$	$(100-54)\ mod\ 256 = 46$
$(105-51)\ mod\ 256 = 55$	$(97-49)\ mod\ 256 = 48$
$(98-49)\ mod\ 256 = 49$	$(99-51)\ mod\ 256 = 48$
$(102-53)\ mod\ 256 = 49$	$(105-57)\ mod\ 256 = 48$

Langkah terakhir menentukan karakter dari indeks plainteks, Indeks karakter yang didapatkan setelah proses dekripsi menggunakan algoritma *OTP* selanjutnya dikonversi dalam bentuk karakter asli. Hasil konversi dapat dilihat pada **Tabel 13.**

Tabel 13. Konversi Indeks Karakter Ke Bentuk Desimal

ASCII	**Plainteks**
54	6
53	5
46	.
53	5
50	2
52	4
46	.
55	7
49	1
49	1
46	.
50	2
45	–
51	3
51	3

49	1
46	.
48	0
48	0
48	0

Dengan demikian hasil dekripsi cipherteks Ëúµ°#╚$FF A╬Ä + ƒ˙SIÊ¬Õ³SP Menggunakan algoritma *One Time Pad* dan operasi *Exclusive-OR* adalah **65.524.711.2-331.000.**

4. Conclusion

Berdasarkan hasil penelitian, dapat diambil beberapa kesimpulan seperti:

1. Proses enkripsi NPWP menggunakan kombinasi algoritma *One Time Pad* dan operasi *Exclusive OR* diawali dengan menentukan kunci yang dibangkitkan secara acak dan pembangkit bilangan acak Blum Blum Shub. Selanjutnya, masukkan indeks karakter dari setiap huruf plaintext ke dalam kunci enkripsi. Kemudian lakukan perhitungan sehingga menghasilkan *cipherteks* Ëúµ°#╚$FF A╬Ä + ƒ˙SIÊ¬Õ³SP.
2. Proses dekripsi NPWP menggunakan kombinasi algoritma One Time Pad dan operasi Exclusive OR diawali dengan mengkonversi cipherteks menggunakan Tabel ASCII. Selanjutnya, masukkan indeks karakter dari setiap huruf plaintext ke dalam kunci dekripsi. Kemudian lakukan perhitungan sehingga menghasilkan *plainteks* 65.524.711.2-331.000.

5. References

Ariyus, Doni. (2008) Pengantar Ilmu Kriptografi: Teori Analisis & Implementasi. Andi. Yogyakarta.

Batten, L. M. (2012). Public Key Cryptography Applications and Attacks. Netherlands: IEEE Press.

Das, A. (2013). Computational Number Theory. Boca Raton: Taylor & Francis Group.

Halim, Abdul, Icuk Rangga Bawono, dan Amin Dara. (2016). Perpajakan, Edisi Kedua. Salemba Empat. Jakarta.

Irawan, W. H. (2014). Pengantar Teori Bilangan. Malang: UIN-Malang Press.

Kumar, D. S., Suneetha, C. H., & Chandrasekhar, A. (2012). A Block Cipher Using Rotation and Logical XOR Operations. IJCSI International Journal of Computer Science Issues, 8(6).142-147.

Munir, Rinaldi. (2006). Kriptografi. Informatika. Bandung.

Munir, Rinaldi. (2019). Kriptografi edisi dua. Informatika. Bandung.

Sanjaya, M. B. (2017). Perancangan dan Implementasi Random Number Blum Blum Shub pada Dynamic Cell Spreading untuk Pengamanan Berkas. Seminar Nasional Multi Disiplin Ilmu.

Sidik, A. P., Efendi, S., dan Suherman, S. (2019). Improving One-Time Pad Algorithm on Shamir's Three-Pass Protocol Scheme by Using RSA and ElGamal Algorithms. Journal of Physics: Conference Series 1235(1).

Stallings, W. (2003). Cryptography and Network Security. New Jersey: Pearson Education.

Trisnawati, T. T., Yurinanda, S., Syafmen, W & Multahadah. C. (2023). Penerapan Algoritma Rivest-Shamir-Adleman (RSA) pada Enkripsi Uniform Resource Locator (URL) Website untuk Keamanan Data. EULER: Jurnal Ilmiah Matematika, Sains dan Teknologi, 11(2). 205-215.

Undang- Undang Nomor 28 tahun 2007 Perubahan Ketiga atas Undang-Undang Nomor 6 Tahun 1983 tentang Ketentuan Umum dan Tata Cara Perpajakan.

https://mediasuara.com/2020/08/18/jangan-gunakan-npwp-orang-lain-atau-faktur-tbts-untuk-kegiatan-usaha-anda/ diakses pada tanggal 10 Maret 2023.