ISSN: 2321-7782 (Online)
Impact Factor: 6.047

Volume 4, Issue 10, October 2016

International Journal of Advance Research in Computer Science and Management Studies

Research Article / Survey Paper / Case Study

Available online at: www.ijarcsms.com

3-dimensional Noise Wave Driven Encryption Algorithm Ver-1(3dNWDEA-1)

Dr. Asoke Nath[1]
Department of Computer Science
St. Xavier's College(Autonomous)
Kolkata, West Bengal – India

Aashijit Mukhopadhyay[2]
Department of Computer Science
St. Xavier's College(Autonomous)
Kolkata, West Bengal – India

Abstract: ***The authors have proposed a new encryption algorithm where double encryption applied on plain text using two different parts of the same secret key. At first the plain text padded with the secret One Time Pad of the key and encrypted using the generated passkeys from the AASN algorithm [3] introduced earlier by the authors. The second encryption is done on the cipher text embedded with the time stamp using the secret One Time Pad. In this algorithm, the padded bits have been stored in a three-dimensional Boolean array and wave has been introduced over it. Primary Wave of a calculated wavelength and starting position has been applied in the vertical direction to find the points where secondary wave can be applied in that particular layer of data. Those points are used to apply secondary waves in the forms concentric circles over the bits and complementing every bit that fall under their trajectory. Ultimately the layered bits are converted into a binary stream and padded with the time stamp given from the AASN algorithm [3]. The bits are now converted into a single two-dimensional data layer and noise wave is applied from both sides of the layer and also from a particular point in between the points with a calculated initial and final radius. These data are calculated from the secret One Time Pad used by the user-receiver pair. The bits are converted into their corresponding ASCII codes and are transmitted through any insecure channel. This technique has been successfully implemented on Android OS to encrypt OTP (One Time Password) used for financial transactions.***

Keywords: NDEA-1; NDEA-2; AASN; Primary Wave; Secondary Wave.

I. INTRODUCTION

The most important and perhaps the most interesting portion of the cryptography might probably lie in securing financial transactions taking place over the internet. OTPs (**O**ne **T**ime **P**asswords) have been used to give the end user an extra level of security but while performing Debit Card transactions, no passwords are required except for the CVV and the OTP which flies unsafe through the network.

The authors have introduced a new technique of double encryption which can prove to be faster than its predecessor algorithms and at times much safer than them. In previous versions of this algorithm, namely NDEA-1 [1] and NDEA-2 [2] has been noise wave had been introduced over the plain text bits stored in a two-dimensional array. In NDEA-1 [1] noise was implemented by scraping out binary windows from the whole two-dimensional data layer and applying noise on it. It used concepts of overwriting to write over the binary data several times by applying noise on the data bits. This form of cryptography technique was enhanced to form NDEA-2 [2]. In NDEA-2 data bits was again stored in a two-dimensional array. Primary wave and Secondary wave was applied to the binary data stored in the array using concepts of constructive and destructive interference. This method was very secured but very slow so it was proposed to be used in the fields where security was of higher priority than that of speed of the algorithm.

In this particular technique, security and speed has been taken care of while developing the algorithm. Here, the plain text is first padded with the secret key entered by the user and then encrypted using the passkeys developed by the AASN algorithm [3]. The passkeys are used to calculate the wavelength of the primary wave and the starting position of the primary wave. The primary wave is applied in the y-z direction towards the three-dimensional data bit layer. The position at which the wave touches the layered binary data are recorded to apply secondary wave. The points through which the wave passes through are used to apply secondary wave in that particular layer. Secondary wave is in the form of concentric circles from the particular point specified from the primary wave traversal. The starting radius of the secondary wave and the ending radius of the secondary wave are given by the passkeys developed by the AASN [3] algorithm. After the application of the waves the 3-D binary array are transformed into a binary stream of bits where the time stamp extracted from the AASN algorithm [3] are embedded together and made ready for the secondary encryption process to take place. In the secondary encryption process the embedded stream is converted into a two-dimensional binary array of width and height calculated from the secret keys entered by the user. Here noise wave is applied in two directions from the starting index of the array and from the ending index of the array. Noise is also applied from a random position from inside the array where the position is determined from the secret key entered by the user. This double encrypted bit array in then transformed into binary stream and then finally converted to their corresponding ASCII codes which can be easily transmitted through an insecure channel without the fear of getting hacked.

II. ALGORITHM

2.1 Governing Equations

Primary Wave Equation:

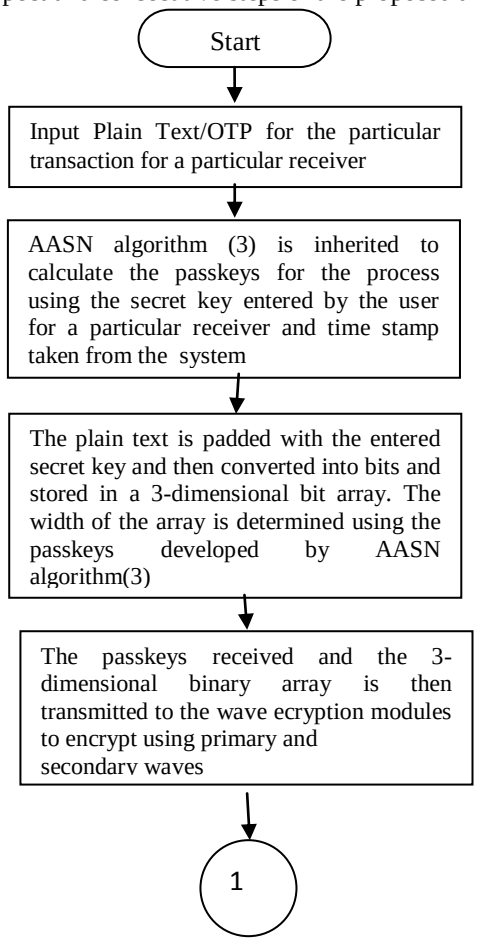
$$Y=A \sin (\omega_0 t + x) \quad (1)$$

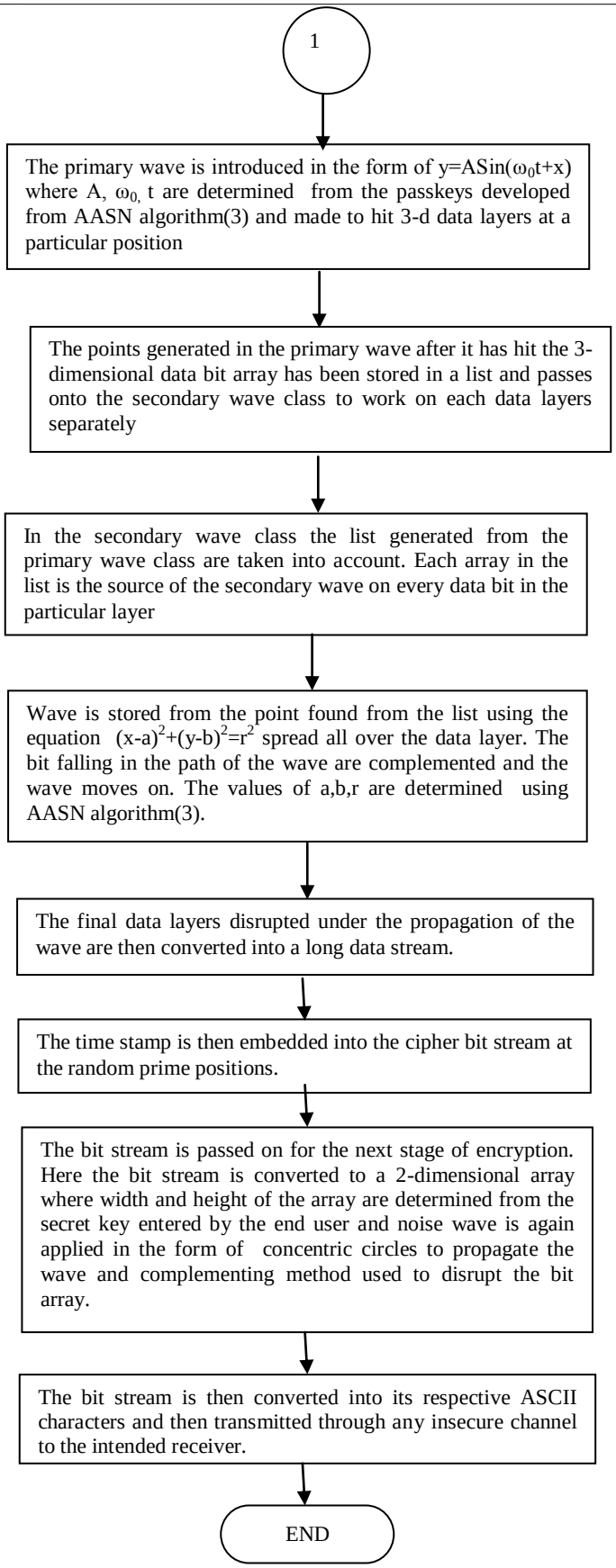
Secondary Wave Equation:

$$(x-a)^2 + (y-b)^2 = r^2 \quad (2)$$

2.2 Block Model of the process

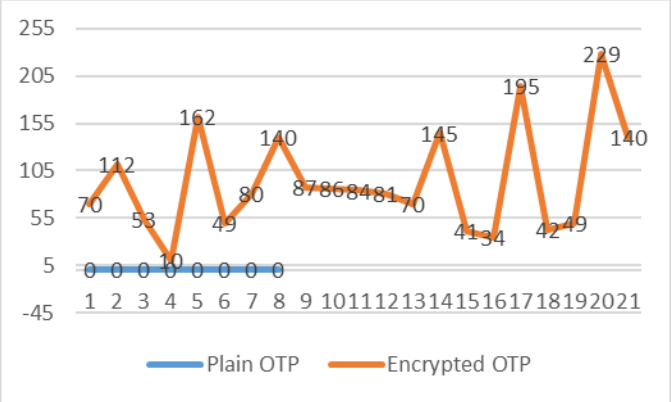
The Block Model describes every aspect and consecutive steps of the proposed algorithm





III. Results and Discussion

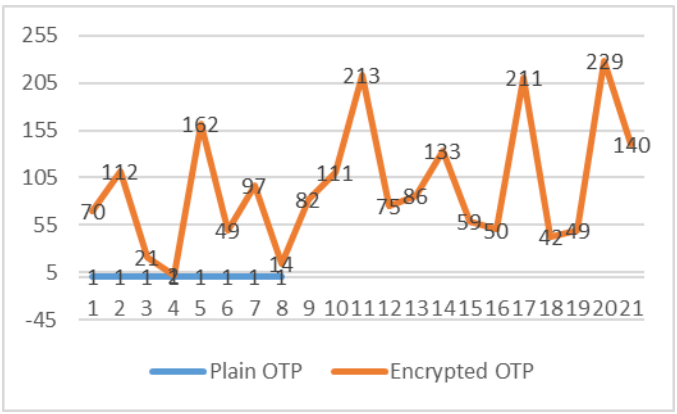
Test Case 01: Encrypting trivial OTP such as OTP=00000000 and Secret Key = abcd



Enter the Secret One_Time_Pad : abcd

Encrypted ASCII Codes: 70,112,53,10,162,49,80,140,87,86,84,81,70,145,41,34,195,42,49,229,140

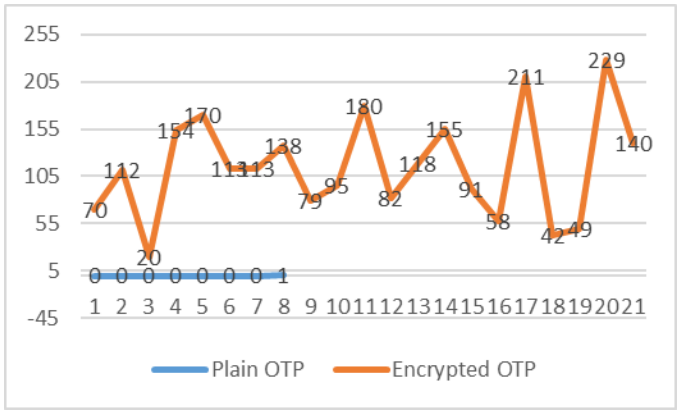
Test Case 02: Using the same secret Key=abcd to encrypt another trivial OTP=11111111



Enter the secret One_Time_Pad : abcd

Encrypted ASCII Codes: 70,112,21,2,162,49,97,14,82,111,213,75,86,133,59,50,211,42,49,229,140

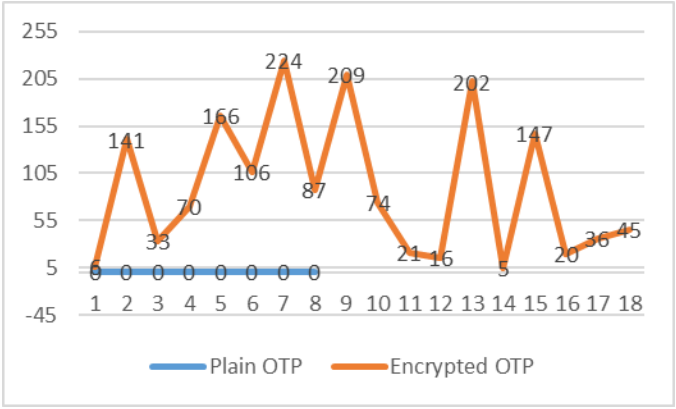
Test Case 03: OTP is changed just by one bit, the change is observed. **OTP=00000001 Secret Key=abcd**



Enter the secret One_Time_Pad : abcd

Encrypted ASCII Codes : 70,112,20,154,170,113,113,138,79,95,180,82,118,155,91,58,211,42,49,229,140

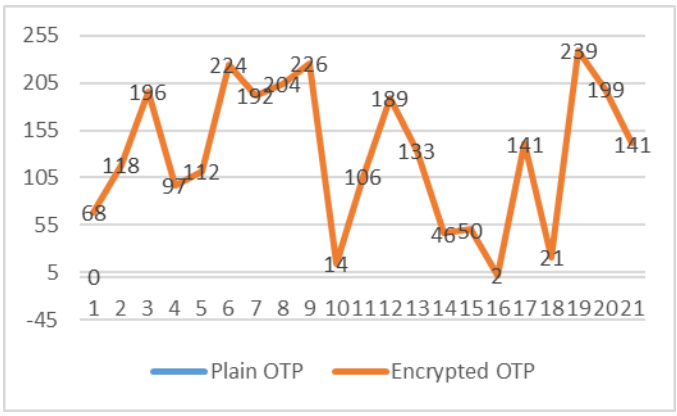
Test Case 04 : Single Letter Secret Key can be also used to encrypt very trivial OTPs, **OTP=00000000 Secret Key=0**



Enter the secret One_Time_Pad : 0

Encrypted ASCII Codes : 6,141,33,70,166,106,224,87,209,74,21,16,202,5,147,20,36,45

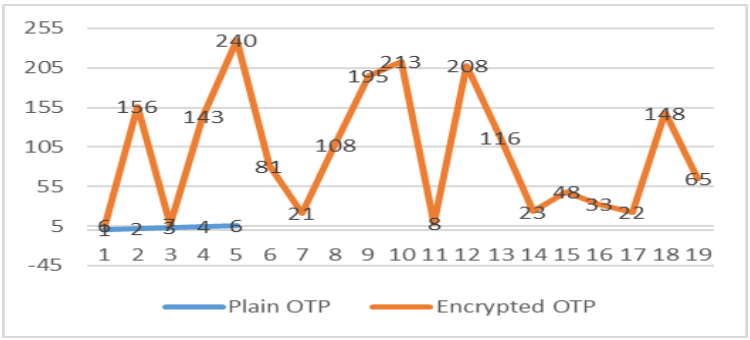
Test Case 05: Using a rather small secret Key to encrypt a single letter **OTP Secret Key="Hello World" OTP=0**



Enter the secret One_Time_Pad : Hello World

Encrypted ASCII Codes : 68,118,196,97,112,224,192,204,226,14,106,189,133,46,50,2,141,21,239,199,141

Test Case 06 : OTP and Secret Key differs just by one bit in the ending position, **Secret Key=12345, OTP = 12346**



Enter the secret One_Time_Pad : 123456

Encrypted ASCII Codes : 6,156,3,143,240,81,21,108,195,213,8,208,116,23,48,33,22,148,65

IV. Conclusion and Future Scope

The algorithm has been tested on various test cases to prove its strength against all known Cryptographic attacks. Embedding the secret key and the time stamp together makes quite a large key for encryption which stands as a barrier against the brute force attack. The algorithm hides single letter frequency, so it cannot be deciphered by statistical attack. The known plain text and known cipher text attacks are also removed from consideration while discussing the shortcoming of the algorithm. Encrypting

same OTP using different passwords not only show different results but also results of different output size. So, it makes deciphering more difficult without knowing the secret key.

Regarding the previous versions, NDEA-1 [1] was also successful in dealing different type of plain text messages and encrypting them and producing random possibilities. In NDEA-2 [2] the resulting possibilities were also random and overall the algorithm was very efficient to curb all forms of cryptographic attacks. But the main problem in the predecessor algorithms were time limit. They required a lot of time to encrypt small data. But the data transmitted was highly secured. In NDEA-3 both the time limit and security has been given equal priority while developing the algorithm. This algorithm has been successfully implemented in MOBILE OS like Android for showing its real world application in encrypting OTPs required for securing financial transactions over the internet.

ACKNOWLEDGEMENT

The authors would like to thank Dr. J. Felix Raj, Principal, St. Xavier's College(Autonomous), Kolkata and the Department of Computer Science of St. Xavier's College (Autonomous), Kolkata for their support in helping the authors to do research in Network Security.

References

1. Noise Driven Encryption Algorithm (NDEA) Version-1: Asoke Nath, Aashijit Mukhopadhyay, Naved Ahmed Tagala, Somnath Saha, International Journal of Innovative Research in Advanced Engineering(IJIRAE) Dec 2015 issue 12, Vol -02J.
2. Noise Driven Encryption Algorithm Version-2 (NDEA – 2): Asoke Nath, Aashijit Mukhopadhyay, Somnath Saha, Naved Ahmed Tagala, Conference: IEEE International Conference CSNT 2016 held in Chitkara University, Chandigarh, March 5-7, 2016, At Chitkara University, Chandigarh
3. AASN algorithm unpublished
4. Asoke Nath, Bidhusundar Samanta, Modern Encryption Standard Ver-V(MES-V), International Journal of Advanced Computer Research(IJACR), Volume-3, Number-3, Issue-11, September 2013, Pages:257264.
5. AsokeNath, Saima Ghosh, Symmetric Key Cryptography using Random Key generator: Meheboob Alam Mallik: "Proceedings of International conference on security and management(SAM "10)" held at Las Vegas, USA July 12-15, 2010), Vol-2, Page: 239-244(2010).
6. DriptoChatterjee, JoyshreeNath, SoumitraMondal, SuvadeepDasgupta and AsokeNath, Advanced Symmetric key Cryptography using extended MSA method: DJSSA symmetric key algorithm: Journal of Computing, Vol 3, Issue-2, Page 66-71,Feb(2011).
7. DriptoChatterjee, JoyshreeNath, SuvadeepDasgupta and AsokeNath, A new Symmetric key Cryptography Algorithm using extended MSA method: DJSA symmetric key algorithm,: Proceedings of IEEE International Conference on Communication Systems and Network Technologies, held at SMVDU(Jammu) 03-06 June,2011, Page-89-94(2011).
8. NeerajKhanna, JoelJames,JoyshreeNath, SayantanChakraborty, AmlanChakrabarti and AsokeNath, New Symmetric key Cryptographic algorithm using combined bit manipulation and MSA encryption algorithm: NJJSAA symmetric key algorithm: Proceedings of IEEE CSNT-2011 held at SMVDU(Jammu) 03-06 June 2011, Page 125-130(2011).
9. Debanjan Das, JoyshreeNath, Megholova Mukherjee, NehaChaudhury and AsokeNath, An Integrated symmetric key cryptography algorithm using generalized vernam cipher method and DJSA method: DJMNA symmetric key algorithm:: Proceedings of IEEE International conference: World Congress WICT-2011 held at Mumbai University 11-14 Dec, 2011, Page No.1203-1208(2011).
10. Trisha Chatterjee, Tamodeep Das, JoyshreeNath, ShayanDey and AsokeNath, Symmetric key cryptosystem using combined cryptographic algorithms-generalized modified vernam cipher method, MSA method and NJJSAA method: TTJSA algorithm –, Proceedings of IEEE International conference: World Congress WICT-2011 t held at Mumbai University 11-14 Dec, 2011, Page No. 1179-1184(2011).

AUTHOR(S) PROFILE

Dr. Asoke Nath, is Associate Professor in the Department of Computer Science, St. Xavier's College(Autonomous), Kolkata, India. Apart from his teaching assignment he is actively envolved in doing research work in Computer Science and Engineering. His research areas include Data eccryption and Cryptography, Steganogarphy, DNA Cryptography, Big data analytics, Data science, Cognitive radio, Green computing, MOOCs, e-learning methodologies, Mathematical modelling of Social Networks, Li-Fi technology and so on. He has already published more than 201 research publications in Journals and Proceedings of International conferences.

Aashijit Mukhopadhyay, born in Kolkata on 8th November, 1994. The author has always been interested in working with systems that are smart and secured. Network security happens to be a passion of the author. The author is doing his hounours course from St. Xavier's College (Autonomous), Kolkata. The author has published few of his papers on Network Security in international journals and conferences. The author has also a nack of implementing every work of his in mobile os like Android. The author is currently busy in implementing database security in Android OS.

