A Review Paper on Encryption Techniques

Sanjeev Kumar Mandal[1], A R Deepti[2]

[1]Research Scholar, Department of Master of Computer Applications, Visvesvaraya Technological University, India.
sanjeev.mandal93@gmail.com
[2]Professor, Department of Computer Science, Indian Academy Degreee College, Bangalore, India.

Abstract : Today's world is depend on internet and its application where transmitting data through network communication via mail, social group, online banking etc., Hence there comes the requirement of securing the information is a must and so cryptography techniques are employed such as symmetric key and asymmetric key techniques. In this paper, few symmetric and asymmetric key encryption techniques are reviewed, compared and tested with cryptography tools to check its security strength.

Keywords: Cryptography, RSA, DES, AES, DSA, ECC and security tools.

1. Introduction

Today's world, is more dependent on network communication when it comes to application and usage of internet in online banking, shopping, social network etc.(Stankovic, 2014), the high the growth in networking technology ends up in interchanging great amount of knowledge. Hence, while transmitting confidential information, there comes the requirement of data security (Elmaghraby & Losavio, 2014). The idea of encryption/decryption algorithms are used where the encryption is scrambling the plain message into cipher text and vice versa for decryption (Hall, 2003). This encoding/decoding is done with the help of a secret key or code. Without the key an intruder cannot able to decrypt the messages (Heckerman, 1995). Hence cryptography plays a vital role in securing the information (Murray, 1995). The cryptography is divided into many types here we see two types namely symmetric and asymmetric cryptography (Chandra et al., 2014). Symmetric key cryptography is one where use of single key for both encryption and decryption (Delfs & Knebl, 2015). Whereas in public key cryptography one key act as public key for encryption and the other is private key to decrypt the message (Pinckney et al., 2017).

Therefore this paper discusses about few encryption mechanisms in section 2 with a comparative study in section 3 and tools have been used to prove whether it is secure or not in section 4 and conclude with future work in section 5.

Thus in this paper the analysis is devoted to some of symmetric and asymmetric-based cryptography (Chandra et al., 2014). Readers fascinated by learning the practicality of contemporary uneven theme are cited. A simplified scenario of Cryptography encryption/decryption is shown in Figure 1.

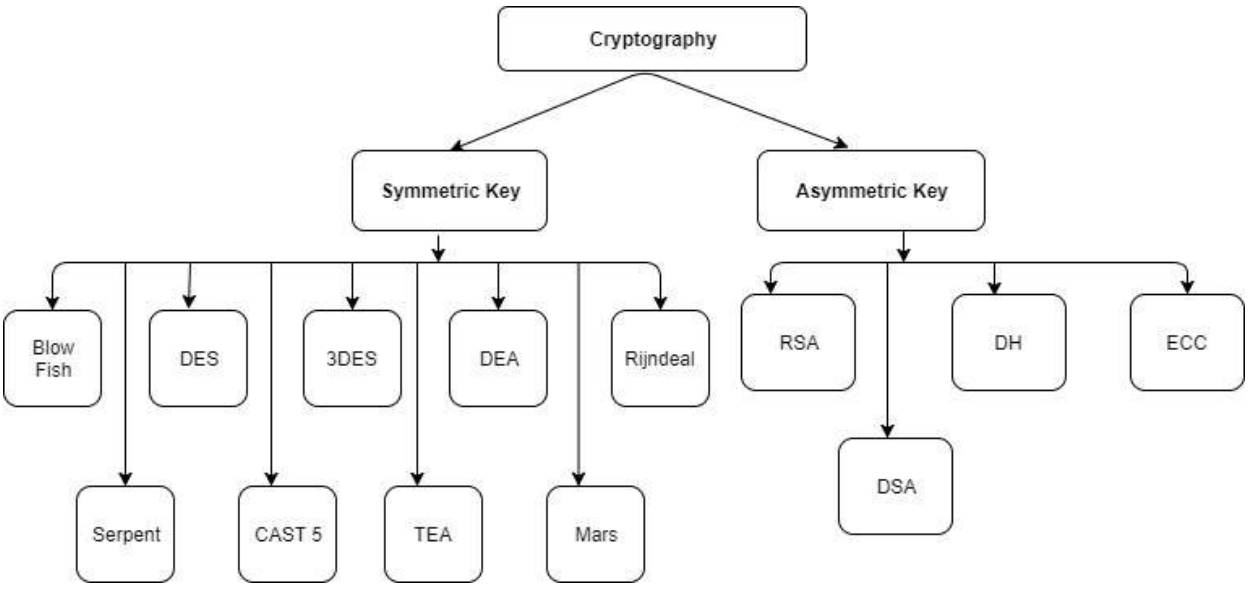

Figure 1: Symmetric and Asymmetric Encryption Method

2. Symmetric key

In the symmetric key cryptography, the same key is used between sender and receiver for the encryption/decryption method (Delfs & Knebl, 2015). Symmetric key cryptosystems are much quicker than the Asymmetric key cryptosystems (Blaze et al., 1997), because of its single key sharing system, where this method have several algorithm like DES (Singh & Supriya, 2013), 3DES (Paar & Pelzl, 2010), BlowFish (Bhanot & Hans, 2015), IDEA (Bhanot & Hans, 2015), TEA (Yee Hunn, Siti, & Binti Idris, 2012), CAST 5 (Bellare &

Tackmann, 2016), Rijndael (Rose & Hawkes, 2003), RC6 (Rivest et al., 1998), Serpent (Courtois & Pieprzyk, 2002), 2 Fish (R.S. & R.D., 2012) and MARS (Burwick & Coppersmith, 1999) which is discussed below.

2.1 DES

DES was developed by IBM in 1977 for protecting data from unauthorized access (Biham & Shamir, 1991). It uses blocks size sixty-four bits and key size fifty-six bits (Biryukov & Cannière, 2006)a. It always operates on blocks of equal size and uses each permutation and substitutions within the rule. It uses sixteen rounds of transposition and substitution to write every cluster of eight (64 bit) plaintext letters and created output from every round one by one (Biryukov & Cannière, 2006). The quantity of rounds is exponentially proportional to the number of our time. Therefore, the number of rounds will increase then the security of the rule will increase exponentially. In general, DES was proven insecure for large companies or governments and it's simpler to not use DES rule. However, for backward compatibility and price of upgrading, the risk of exposure, DES is extremely liable to linear scientific discipline attacks. Weak keys also are a great issue. DES is additionally exposed to brute force attack.

2.2 Triple DES (3DES)

Triple DES is designed to improve drawback in DES in Nov 1998 (Bhanot & Hans, 2015). It uses 3 64-bit keys and an overall key length of 192 bits. It supported the thought of Feistel structure and additionally contains eight S-boxes. The procedure for encoding is precisely the identical as DES however this method is recurrent thrice while not dynamic the initial structure of DES algorithmic program. It operates as inscribe, decrypt/encrypt. This procedure for decrypting is that the same because of the procedure for encoding, except it is accepted the same as reverse method (S & Muruganandam A, 2014). 3DES is exposed to differential and related-key attacks in addition to that it is prone to the bound variation of meet-in-the-middle attack (Ahmad et al., 2010). 3DES offers a high level of security compared with DES and continues to be utilized by the U.S.

2.3 Blow Fish Algorithm

Blow Fish algorithmic is a replacement for DES or IDEA algorithm (El-etriby, Mohamed, & Abdul-kader, 2012). It is symmetric key cryptography which uses a block cipher method with variable length key which uses 32 bits to 448 bits, which is use for domestic and exportable use. It supported sixteen round feistel cipher network that uses the big key size. The key size is larger because it is difficult to break the cipher text within the blowfish algorithmic rule. Blow Fish has some categories of weak keys. Four rounds of blowfish are exposed to ordinal order differential attacks (Heys, 2002). So, reliable of Blowfish is questionable because of the big range of weak keys. Blowfish's security lies in its variable key size (128-448 bits) providing a high level of security. Blowfish is defendable against differential related-key attacks, since as of the key involves several spherical keys that are significantly freelance, creating such attacks very sophisticated or impossible. Such autonomy is extremely desirable (Bhanot & Hans, 2015).

2.4 IDEA

International encryption algorithm (IDEA) was developed by James L. Massey and Xuejia Lai (Zurich, Switzerland) in 1990 (Biham, Biryukov, & Shamir, 1999). It's an even key formula supported the conception of substitution-permutation structure. It fairly quick, thought-about secure, and is additional proof against each linear and differential analysis. it's a block cipher that uses a sixty-four bit plain text with eight rounds and a key length of 128-bit permuted into fifty-two sub- keys every of 128- bits. It doesn't contain S- boxes and the same formula is employed in reversed for secret writing. The plan includes a robust resistance against differential science beneath bound hypothesis. The plan makes use of multiple cluster operations to extend its strength against most acquainted attacks (Biham, Biryukov, & Shamir, 1999). It consists of 128-bit key size creating it as a robust security formula. No weaknesses relating linear or algebra attacks have however been reported (We, 2006). The most effective attack that applies to any or all keys will break which is reduced to six rounds.

2.5 TEA

Tiny Encryption Algorithm (TEA) was designed by David Wheeler and Roger Needham of the Cambridge Computer Laboratory in 1994 (Wheeler & Needham, 2012). It is known for its simple structure and easy implementation, typically a few lines of code. It is also a Feistel structured symmetric key algorithm. It is a block cipher that uses a 64-bit plain text with 64 rounds and a key length of 128-bit with variable rounds having 32 cycles. It does not contain S- boxes and the same algorithm is used in reverse for decryption. TEA algorithm offers the same security level as that of IDEA. TEA is also susceptible to a related-key attack. Because of these weaknesses, the XTEA cipher was designed.

2.6 AES (Rijndael)

Rijndael was developed by Joan Daemen and Vincent Rijmen in Oct 2000 declared by the National Institute of Standards and Technology (Daemen, Rijmen, & Leuven, 1999). Rijndael victimization variable key size is an extraordinarily quick and compact cipher. The AES could be a block cipher that uses a 128-bit plain text with variable ten, 12 or fourteen rounds and a variable key length of 128, 192, 256 bit permuted into ten sub- keys every of 128, 192, 256-bit length severally. It solely contains one S- box and the same algorithmic rule is employed in reversed for cryptography.

AES is additionally a cruciform key algorithmic rule supported the Feistel structure. Security of Rijndael depends on its variable nature key size permitting up to a key size of 256-bit, to produce resistance against sure future attacks (collision attacks and potential quantum computing algorithms) (Paar & Pelzl, 2009). General attacks of Irondale are sq. Attack, Improved sq. Attack, not possible Differential Attack and Reversed Key Schedule Attack, however, none of the attacks were much attainable.

2.7 RSA

The RSA algorithm is the basis of a cryptosystem, a collection of cryptanalytic algorithms that are used for specific security services or functions which allows public key coding and is wide wont to secure sensitive information, significantly once it's being sent over an associate insecure network like the web (Callas, 2003). RSA algorithm is asymmetric cryptography algorithm which uses of 2 key namely public key and private key, from the name itself it shows public key which means for all use private means for every user will be having one private key which will differ from everyone.

An example of asymmetric cryptography:

1. A consumer (for example browser) sends its public key to the server and requests for a few information.
2. The server encrypts the information using the client's public key and sends the encrypted data.
3. The client receives this information and decrypts it.

Since this is often uneven, no one else except browser will decode the information whether or not a 3rd party has the public key of the browser.

2.8 DSA

Digital signatures are the public-key primitives of message authentication (Arya, Aswal, & Kumar, 2012). Within the physical world, it is common to use written signatures on handwritten or written messages. They're accustomed bind mortal to the message. Similarly, a digital signature could be a technique that binds a person/entity to digital information. This binding will be severally verified by receiver similarly as any third party. Digital signature could be a cryptanalytic worth that's calculated from the info and a secret key familiar solely by the signer. In the planet, the receiver of the message wants assurance that the message belongs to the sender and he mustn't be ready to repudiate the origination of that message. This demand is incredibly crucial in business applications since the chance of a dispute over changed information is incredibly high.

2.9 ECC

Elliptical curve cryptography (ECC) is a public key encoding technique supported elliptic curve theory that can be used to produce quicker, smaller, and more economical cryptanalytic keys. ECC generates keys through the properties of the elliptic curve equation rather than the standard methodology of generation because of the product of terribly giant prime numbers (Al-Hamdani, 2011). The technology is employed in conjunction with most public key coding ways, like RSA (Callas, 2003), and Diffie-Hellman (Callas, 2003). in keeping with some researchers, ECC will yield grade of security with a one 64-bit key that alternative systems need a 1,024-bit key to attain. As a result of ECC helps to ascertain equivalent security with lower computing power and battery resource usage, it's turning into wide used for mobile applications.

2.10 Diffie-Hellman

Diffie-Hellman key exchange, also known as an exponential key exchange, is a methodology of digital cryptography that uses numbers raised to specific powers to supply cryptography keys on the idea of parts that are ne'er directly transmitted, creating the task of a would-be code breaker mathematically overwhelming (Murray, 1995).

2.11 RC5

RC5 is a parallel key block coding formula designed by Ron Rivest in 1994. it's notable for being easy, quick (on account of using only primitive operations like XOR, shift, etc.) and consumes less memory (Rivest et al., 1998). RC5 is a block cipher and addresses 2-word blocks at a time. Depending on input plain text block size, the range of rounds and key size, varied instances of RC5 are often outlined and every instance is denoted as RC5-w/r/b wherever w = word size in bits, r = number of rounds and b = key size in bytes.

3. Comparative Study of Different Encryption Method

Input Size (Kilobyte)	Results of Encryption	Average Time	Encryption Speed
46, 104, 328, 905, 5202	27, 45, 75, 257, 987	310.2	Fast
46, 104, 328, 905, 5202	54, 87, 157, 270, 1108	310.2	Fast
46, 104, 328, 905, 5202	55, 92, 167, 267, 1200	378.9	Fast
46, 104, 328, 905, 5202	40, 47, 87, 123, 157	96.3	Fast
46, 104, 328, 905.	56, 92, 176, 306.	489.6	Fast
46, 104, 328, 905, 5202	20, 27, 37, 113, 152	76.3	Slow
46, 104, 328, 905, 5202	55, 92, 167, 267, 1200	378.9	Fast
46, 104, 328, 905, 5202	42, 63, 110, 152, 765	247.1	Fast
46, 104, 328, 905, 5202	56, 92, 176, 306, 1507	247.1	Fast
46, 104, 328, 905, 5202	20, 27, 37, 113, 152	76.3	Fast
46, 104, 328, 905, 5202	55, 92, 167, 267, 1200	378.9	Fast
46, 104, 328, 905, 5202	27, 45, 75, 257, 987	310.2	Fast

Encryption Algorithm	Algorithm Structure	Rounds	Key Size (bits)	Plain Text / Cipher Text Length	Cryptanalysis	Block Size (bits)	Time to crack
Data Encryption Standard (DES)	Feistel Structure	16 Rounds (Substitu	56-bit key	64 bits	Differential Cryptana	64-bit block	$7.2 * 10^{16}$
Triple Data Encryption Standard	Feistel Structure	48 Rounds	112\168	64 bits	Meet in the Middle	64-bit block	$1.01 * 10^{18}$
Advanced Encryption Standard (AES)	Feistel Structure	10/12/14 Rounds (SubByt	128/192/256	128 bits	Brute-force	128-bit block	$3.4 * 10^{38}$/ $6.2 * 10^{57}$/ $1.1 * 10^{77}$
Blow Fish	Feistel Structure	16 Rounds	32-448	64 bits	Birthday attack like http	64-bit	$1.01 * 10^{18}$
RC2	Feistel	18 Rounds	40-1024	128 bits	Related Key	64-bit	$1.01 * 10^{18}$
IDEA	Substitution - Permutatio	8	128	64 bits	Bicliques attack	64-bit	$1.01 * 10^{16}$
Rijndael	Feistel	10,12,14	128,192,256	128 bits	Chosen-plain, Known	128-bit	$3.4 * 10^{38}$/ $6.2 * 10^{57}$
RC5	Feistel	12 Rounds	128/192/256	128 bits	Differential attack	64/128-bit	$3.4 * 10^{38}$/ $6.2 * 10^{57}$/ $1.1 * 10^{77}$
RSA	N/A	N/A	>1024 bits	N/A	Timing Attacks	64/128-bit	$3.4 * 10^{38}$/ $6.2 * 10^{57}$/ $1.1 * 10^{77}$
DSA	N/A	N/A	1024	N/A	Key-only known message	N/A	$3.4 * 10^{38}$/ $6.2 * 10^{57}$/ $1.1 * 10^{77}$
ECC	N/A	N/A	1024	N/A	Key-only known message	N/A	$3.4 * 10^{38}$/ $6.2 * 10^{57}$/ $1.1 * 10^{77}$
Diffie-Hellman	N/A	N/A	Key Exchange management	N/A	Eavesdropping		$3.4 * 10^{38}$/ $6.2 * 10^{57}$/ $1.1 * 10^{77}$

4. Analyze the Encryption method in the Encryption tab

From the comparative study, a real-time Symmetric and Asymmetric method are selected like WhatsApp, Http, Https and Https-SSL. For Http and Https IBM Log-in system has been selected and it has been tested in JSKEY tool and proved to be insecure against File Backup Check, Directory with write permission enabled, possible sensitive directory, etc. which will help intruder to create a test file in the directory with the help of web scanner, shown in Figure 2

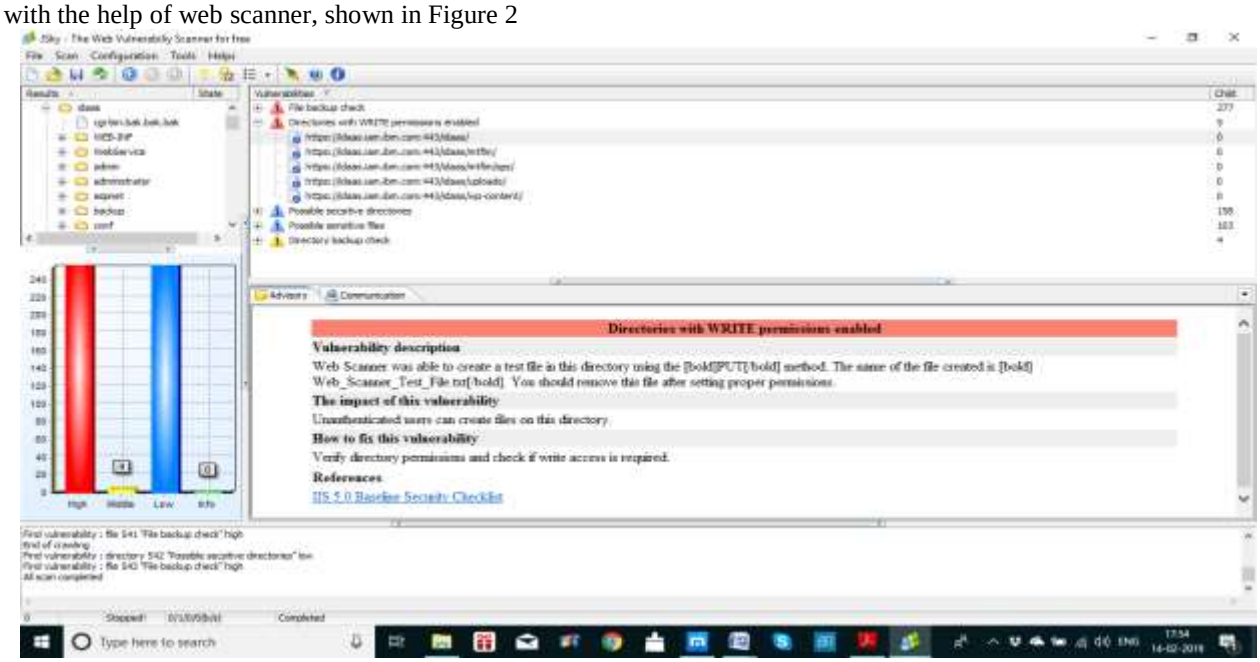

Figure 2: IBM Log-in valunability test with JSKEY Tool

To test HTTPS-SSL Encryption method PAYTM Bank has been used which is having all possible data from the customer with account/wallet. Once again we used JSKEY tool and prove that even Paytm is not secure against SQL injection attack and possible sensitive directory attack is shown in Figure 3.

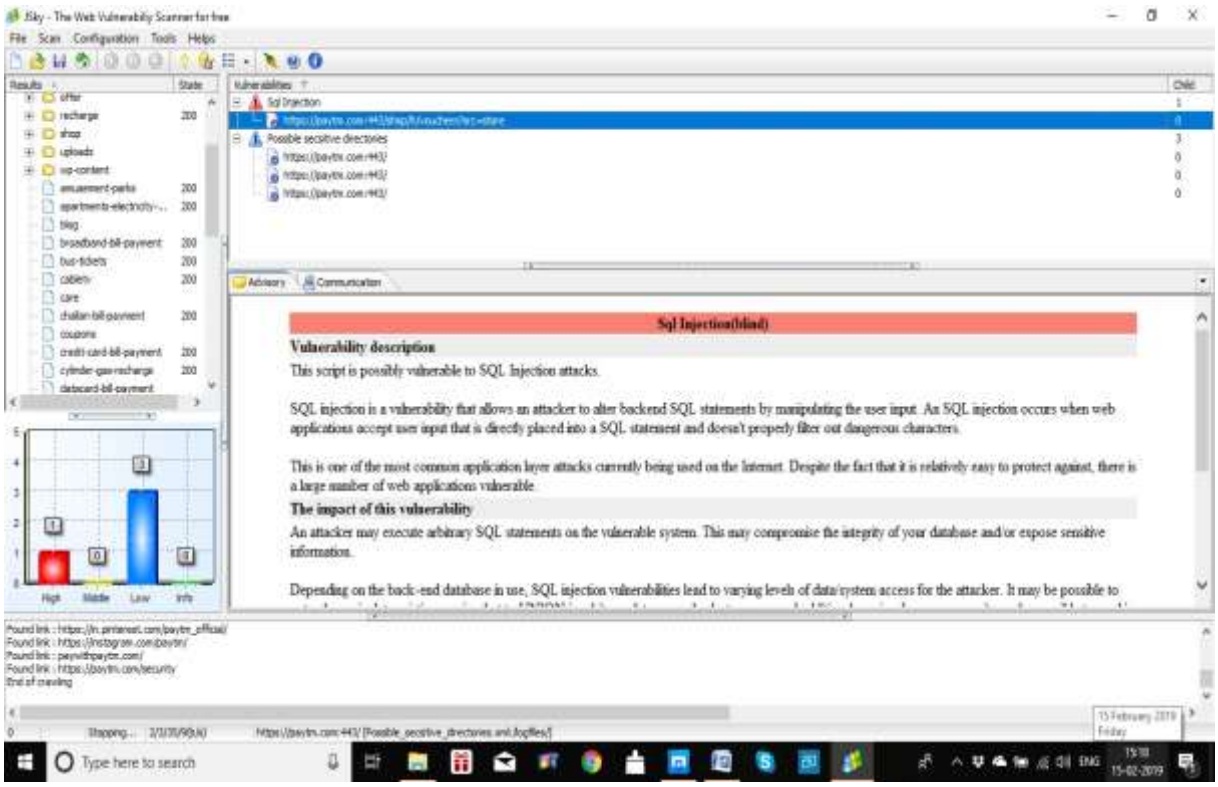

Figure 3: PAYTM is tested in JSKEY Tool

Thus, from the above test, it is proved that these protocols should be improved in order to improve their security.

5. Conclusion and Future Work

In this paper, a general approach of the encryption method is described. Then, a comparative study is made between the encryption algorithms to check its security complexity. Furthermore, the analysis is done in order with the help of a tool to provide a satisfying security level in terms of data transmission and time. Since the research on data encryption is a relatively young area, the number of new formulate encryption is increasing as long as many attacks are appearing so the direction of the future research work is about verifying their efficiency.

Reference

[1] Ahmad, S., beg, D. M. R., Ahmad, J., & Barua, N. 2010. Meet In The Middle Attack: A Cryptanalysis Approach. International Journal of Computer Applications.
[2] Al-Hamdani, W. A. 2011. Elliptic curve for data protection. In: Proceedings of the 2011 Information Security Curriculum Development Conference on - InfoSecCD '11.
[3] Arya, P. K., Aswal, M. S., & Kumar, V. 2012. Comparative Study of Asymmetric Key Cryptographic Algorithms. International Journal of Computer Science & Communication Networks,.
[4] Bellare, M., & Tackmann, B. 2016. The multi-user security of authenticated encryption: AES-GCM in TLS 1.3. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics).
[5] Bhanot, R., & Hans, R. 2015. A review and comparative analysis of various encryption algorithms. International Journal of Security and its Applications.
[6] Biham, E., Biryukov, A., & Shamir, A. 1999. Miss in the middle attacks on IDEA and khufu. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics).
[7] Biham, E., & Shamir, A. 1991. Differential cryptanalysis of DES-like cryptosystems. Journal of Cryptology.
[8] Biryukov, A., & Cannière, C. 2006. Data encryption standard (DES). In: Encyclopedia of Cryptography and Security.
[9] Blaze, M., Diffie, W., Rivest, R., & Schneier, B. 1997. Minimal key lengths for symmetric ciphers to provide adequate commercial security, January 1996. Online at http://www. ….
[10] Burwick, C., & Coppersmith, D. 1999. The Mars Encryption Algorithm. NIST AES Proposal.
[11] Callas, J. 2003. An Introduction to Cryptography. Information Systems Control Journal.
[12] Chandra, S., Paira, S., Alam, S. S., & Sanyal, G. 2014. A comparative survey of symmetric and asymmetric key cryptography. In: 2014 International Conference on Electronics, Communication and Computational Engineering, ICECCE.
[13] Courtois, N. T., & Pieprzyk, J. 2002. Cryptanalysis of block ciphers with overdefined systems of equations. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics).
[14] Daemen, J., Rijmen, V., & Leuven, K. U. 1999. AES Proposal : Rijndael. Complexity.
[15] Delfs, H., & Knebl, H. 2015. Symmetric-key cryptography. In: Information Security and Cryptography.
[16] El-etriby, S., Mohamed, E., & Abdul-kader, H. 2012. Modern Encryption Techniques for Cloud Computing. International

Conference on Communications and Information Technology (ICCIT).
[17] Elmaghraby, A. S., & Losavio, M. M. 2014. Cyber security challenges in smart cities: Safety, security and privacy. Journal of Advanced Research.
[18] Hall, S. 2003. Encoding/decoding. In: Culture, Media, Language: Working Papers in Cultural Studies, 1972-79.
[19] Heckerman, D. 1995. A tutorial on learning Bayesian networks. Technical Report MSR-TR-95-6.
[20] Heys, H. M. 2002. A tutorial on linear and differential cryptanalysis. Cryptologia.
[21] Murray, W. H. 1995. Modern Cryptography. Information Systems Security.
[22] Paar, C., & Pelzl, J. 2009. The Advanced Encryption Standard (AES). In: Understanding Cryptography.
[23] Paar, C., & Pelzl, J. 2010. Understanding Cryptography – A Textbook for Students and Practitioners. Understanding Cryptography – A Textbook for Students and Practitioners.
[24] Pinckney, N., Harris, D. M., Jiang, N., Kelley, K., Antao, S., & Sousa, L. 2017. Public key cryptography. In: Circuits and Systems for Security and Privacy.
[25] R.S., I., & R.D., K. 2012. Alzheimer's diet modification: A web-based nutrition tracking system for patient management and outcomes research. Journal of Nutrition, Health and Aging.
[26] Rivest, R. L., Robshaw, M. J. B., Sidney, R., & Yin, Y. L. 1998. The RC6 block cipher. NIST AES Proposal.
[27] Rose, G. G., & Hawkes, P. 2003. Turing: A Fast Stream Cipher. Fast Software Encryption, 10th International Workshop, FSE 2003, Lund, Sweden, February 24-26, 2003, Revised Papers.
[28] S, K., & Muruganandam A. 2014. Data Encryption and Decryption by Using Triple DES and Performance Analysis of Crypto System. International Journal of Scientific Engineering and Research (IJSER) www.ijser.in ISSN (Online.
[29] Singh, G., & Supriya, S. 2013. A Study of Encryption Algorithms (RSA, DES, 3DES and AES) for Information Security. International Journal of Computer Applications, 67: 33–38.
[30] Stankovic, J. A. 2014. Research directions for the internet of things. IEEE Internet of Things Journal.
[31] We, P. 2006. Lecture Notes 1. October.
[32] Wheeler, D. J., & Needham, R. M. 2012. TEA, a tiny encryption algorithm.
[33] Yee Hunn, S. A., Siti, S. Z., & Binti Idris, N. 2012. The development of Tiny Encryption Algorithm (TEA) crypto-core for mobile systems. In: International Conference on Electronic Devices, Systems, and Applications.