Semiconductor Optoelectronics
ISSN: 1001-5868

ANALYSIS OF PROTOCOL IN IOT & INTEGRATION OF NANOTECHNOLOGY: A PARADIGMS SHIFT

Pardeep Kumar
Research Scholar, ECE Department, IKG PTU, Jalandhar
sharmaashupk@gmail.com

Dr. Amit Gupta
Assistant Professor, ECE Department, IKGPTU, Jalandhar
amitguptaptu@gmail.com

Abstract—The Internet of Things (IoT) environment has ushered in a new era of nanotechnology and information, leveraging cloud services for big data analysis, and making use of numerous devices like smart cards, RFID tags, various types of sensors, and industrial operations controllers with limited resources. The upcoming devices are adopting nanotechnology for the applications like sensing, efficient power consumption, and smart processing of the information signal. The machines/sensor nodes are susceptible to a variety of fresh malware and other developing dangers. For optimal deployment of sensitive data, & to initiate the exporting of the data to a Cloud Service Provider (CSP) server, the data owner protects the information messages [1]. One of the best methods for protecting such IoT applications is lightweight cryptography techniques. Data will be hidden using cryptography, which ensures that all data transmissions are safe, correct, authenticated, and allowed, by making it impossible to extract any key information patterns [2]. A secure communication method that offers increased security, precision, and efficiency requires careful consideration while choosing a cryptographic algorithm. The most popular symmetric encryption algorithms, such as Data Encryption Standard (DES), Triple Data Encryption Standard (3DES), Blowfish, Advanced Encryption Standard (AES), and an asymmetric encryption Algorithm RSA, are examined in this study for their security-related features. The effectiveness of the right encryption method was evaluated based on a variety of metrics and criteria that best fit the needs of the customer.

Keywords: —Nan sensors; Cryptography; Encryption; Decryption; Data Encryption Standard (DES); Triple Data Encryption Standard (3DES); Blowfish; Advanced Encryption Standard (AES), Rivest-Shamir-Aldeman (RSA).

I. INTRODUCTION

The three components that evolves the Internet of Things—interactions [IP-P] between people and people, interactions [IP-T] between people and things, and interactions [IT-T] between things and things[3]. IoT is nothing more than an efficient network of physical things that can communicate with one another online without human involvement [4]. On small devices, traditional cryptographic methods are challenging to implement, so the managements of

different features and trade-offs between security, cost, performance have been taken care by designers of lightweight ciphers. The primary issue in IoT applications lies with protecting users' sensitive data with such low-end, battery-operated devices, so the design needs to be compact, energy-efficient, and fast enough [5].

The design of cryptographic algorithms is cost-effective, safe, requires little memory, simple to implement to operate on multi platforms. Many conventional cryptographic protocols have a challenging trade-off between provisions of security, performance of algorithms, and IoT resource requirements .The applications include, among others, smartcard, smart houses, healthcare monitoring, consumer electronics, environmental care & monitoring, Industrial control &precautions, modernisation of agricultural process ,radio-frequency identification devices (RFID) for exclusive identification [6],[7]& using wireless sensor node & networks for exchange & acquisition [6], [8].

When sending private information to a server hosted by a Cloud Service Provider (CSP), it is best practise for the sending party to encrypt the information first [9]. Data confidentiality is a major consideration in cloud's user data protection. Among other approaches for cryptography, deterministic and probabilistic encryption techniques have been proposed [10, 11]. The plaintext can be encrypted using possible keys that remain producing the same cipher text using deterministic encryption, but this process must be repeated on multiple occasions. Probabilistic encryption is safer than deterministic encryption because it makes it more difficult for cryptanalysts to attack and figure out private information about plaintext from cipher text and the related key. Numerous cipher texts encrypted with various keys make up the plaintext in probabilistic encryption. Because some cybercriminal or intruder having access to the algorithm, may decode or alter any message that was encrypted using a keyless cryptosystem, these ways are usually less secure than key-based systems like Caesar cipher [12].Fig. (1) shows how cryptographic algorithms are divided into the two main categories of keyless and key-based cryptosystems.

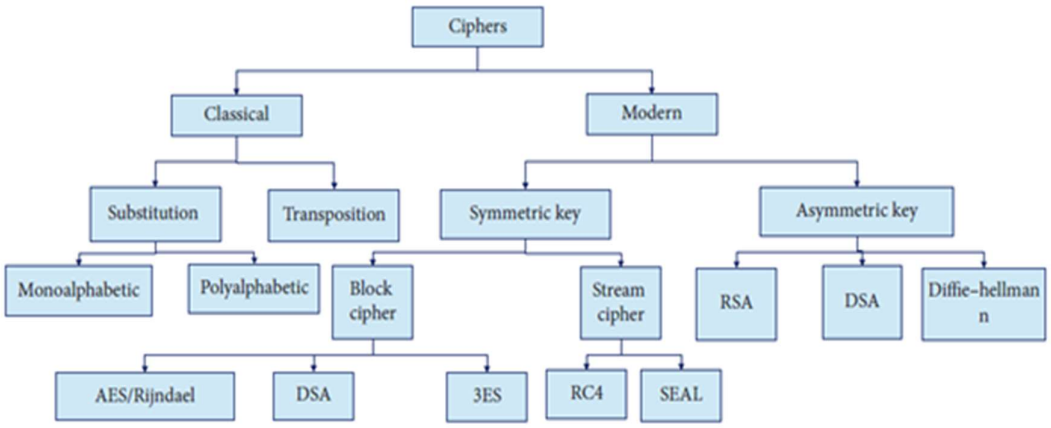


Figure 1: Classification of encryption algorithms

The key-based cryptosystem may be further separated into two ways: (a). symmetric ciphers (secret key) & (b). asymmetric ciphers (public key) [13].

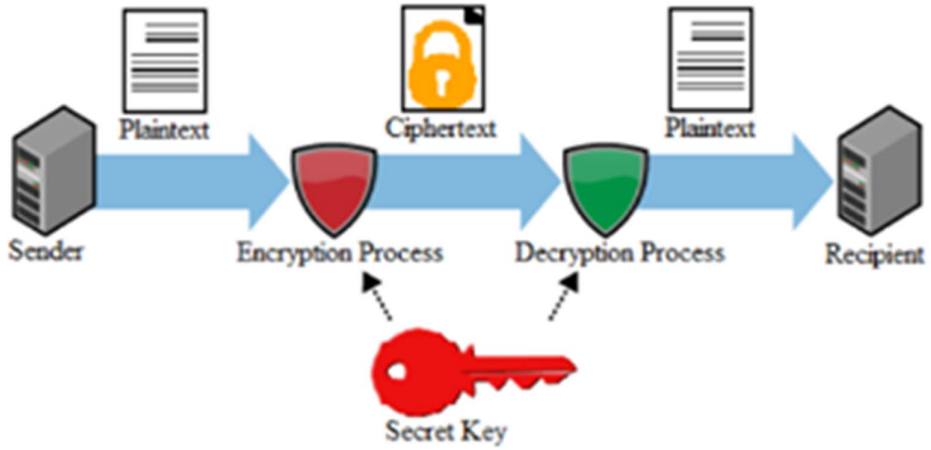


Figure 2:Symmetric Block Cipher

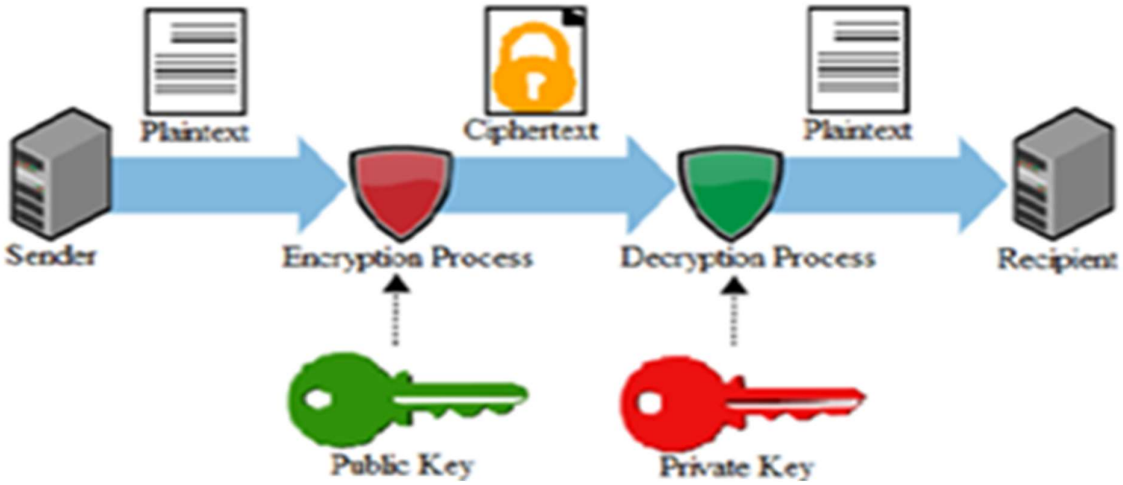


Figure 3: Asymmetric key

Fig. (2) comprised of five main components of Symmetric Block Cipher makes use of secret key schedule algorithm for the effective & secure communication exchange between interested sender & recipient. However, asymmetric algorithms utilise an individual private key to encrypt the message and data decoding [11]. In terms of computing, they can manage enormous data.

The plaintext employing fixed-length blocks of symmetric algorithms like AES [14]. The simulation outcomes and the size of sampled data text files have led to the conclusion that the AES performs better than the RSA and Data Encryption Standard (DES) algorithms[15]. The most used cryptographic primitives are symmetric block techniques. For creation of hash functions[16], the development of pseudorandom (PR) fundamental primitives' sequences, and other operations in addition to ensuring secrecy through encryption in cloud computing [14], [16], [17].

Public and private keys are utilised in asymmetric algorithms, which are further separated into primary key and secondary key configurations. The public key is issued to each participant for initiating encoding process, whereas the private key is kept secret from others and is useful for decoding [18]. Asymmetric utilities like (RSA), and elliptic curve-based cryptography are a

few examples of an asymmetric algorithms [19]. They give more security protections but are pricier and need more computational labour. If the desired key is cracked / leaked, then the desired data can be obtained/ altered by the attacker [20]. Different architectures for creating the blocks of the various symmetric Key Block ciphers are utilised in symmetric key cryptographic ciphers [21]. An algorithm's speed and latency correspond to the time complexity of software primitive, whereas Random Access Memory (RAM) and other storage tends to space complexity, required for algorithm.

Another crucial consideration is that for nodes & devices, like RFID the device's power consumption, namely its low average power consumption and tolerable peak power consumption, need on a battery, so ultra-portable devices are constrained in their ability to perform. Traditional encryption algorithms are inadequate for IoT devices because they waste memory and energy on low-resource devices [22].

II. NANOSENSORS IN IOT

The properties of materials are totally changed when we move towards the nanomaterials from the bulk materials. Nanomaterials have the capability to solve the problems like energy conservation & efficient device formation, due to their remarkable properties of quantum confinement effect and large surface are to volume ratio [36]. The properties like optical, mechanical, Electrical & Electronics properties become drastically changed due to large number of atoms are available at the surface. The use of nanotechnology in the devices will enhance their performance in terms of efficiency and efficacy.

The variety of Nano-sensors and their sensitivity, information extraction accuracy, selectivity, and stability of sensors are remarkable. Nano-sensors also useful for real-time measurements, multi-parameter sensing, analysis & support the direct reading of signal detection[37], [38]. Nanostructure devices have solved many problems that the world is facing in realization of IoT related environment.

The functionality of gadgets is depending on the multiple factors like efficiency, accuracy, and sensitivity. Nanosensors consume less power to operate for longer durations with enhanced reliability. IoT with nanodevice & Nanosensors become an energy –efficient network with a reduced &affordable cost application[39,40]. Nanotechnology has opened the path for affordable and growing IoT projects. Graphene based Nanosensors are highly sensitive towards any type of changes in the measurement parameter of IoT nodes.

Author [41]explained that 0D, 1D & 2D based nanomaterials can be used for making energy efficient devices. IoT nanodevices comprised of nanoantenna for transceivers, nano processors for sensor nodes and nanobatteries for power requirements. However, Nanosensors are restricted within IoT endpoints.

Nanowires based sensors exhibits better &fast performance as compared to NH3 sensor [42]. For duplex method of communication for Internet of Things devices, carbon nanotubes (CNTs)-based nanoantenna's could be preferred to graphene-based ones. These sensors are employed in electro analytical applications[43].Carbide based nanoantenna works without any external circuit and it get directly sprayed on any substrate and device and it work as a smart IoT device [49].

Graphene (2D) based sensors shows high conductive nature along with better chemical & electrochemical stability [43] . CNT bundle-based patch antennas have reduced weight & enhanced flexibility as compared to metal structures, making them suitable for wearable's

[44].CNTs & oxide nanoparticles like lithium titanate oxide nanoparticles, TiO2 V2O5 etc, have high density of states, reduced self-discharge rate along with fast charging capabilities making them useful to replace currently used Lithium-ion batteries [36], [45].

[46]The author explained that silicon-based processors may be replaced with programmable carbon nanotube processors, boosting the capability of IoT processors.Nanosensors have good sensitivity, short response time, wide linear range, and long-term stability [47].Optical based sensors have huge scope for application in IoT sectors[48].

III. IOT CHALLENGES

Sensors gather enormous data before passing it to the clouds and they are highly concern about the sensitivity of confidential users' data, privacy and security of devices with appropriate degrees of encryption mechanism, cyber security procedures. Miniaturization, of sensor modules devices with nanotechnology could help in IoT enhancement by forming nanonetworks. To meet all the requirement IoT and their commercialization in the diverse field including healthcare, the future of IoT is IoNT.

IV. STRUCTURAL ANALYSIS

Block ciphers are typically divided into two distinct categories based on their internal structure: Substitution Permutation Networks (SPNs) and Feistel based networks [23] are two main categories, based on internal structure, resulting in the design of Block ciphers; however, based on structure, some sources classify block ciphers into five distinct categories. A few Feistel ciphers have some concerns about its security flaws and SPN ciphers don't, SPN is a more desirable rival than Feistel in the field of lightweight cryptography [24].

Among other structures, Substitution permutation networks (SPNs), Feistel networks, Add-rotate-XOR (ARX [25]), Non-linear feedback shift register (NLFSR) based, and hybrid versions of different structures, are some of the core structures utilised in the many blocks ciphering mechanism [3] that have been implemented with each passing day. The most well-known ciphers AES utilise the most well-known, SPN-base structure [26], [2] DES along with its enhanced version called as Triple-DES, however the most well-known Feistel type ciphers; Kee Loq, the most recognised NLFSR-based cipher; and the Hummingbird belongs to family of hybrid ciphers, [27], [28].

In IoT uses cloud services, so to send information securely and reliably, a minimal cryptographic framework, based on industry standards is required.

The Feistel network and the Substitution-permutation network are based on the concept of symmetric key structures. The Fiestel network's block encryption and decryption procedures are substantially identical [21], with the exception of the necessity for key schedule reversal. The iteration used in the Fiestel network encryption is a hallmark of this method. Data Encryption Standard (DES), its upgraded Triple Data Encryption Standard (3DES) version along- with Advanced Encryption Standard (AES) &Blowfish, have design based on symmetric ciphers [20]. There are many different kinds of asymmetric algorithms. Some examples are the Diffie-Hellman algorithm, the RSA method, and the techniques used to create digital signatures. The memory (RAM) requirements, flexibility of key, Central Processor (CPU) usage time, and encryption /decryption performance of four algorithms are compared. Given their prevalence and ease of implementation, these four algorithms were selected for use in most recent cryptographic systems.

Data Encryption Standard (DES): IBM's DES algorithm, which uses data encryption, was

developed in 1972 and chosen as the standard by the United States government. DES uses 56-bit and 64-bit blocks length, making it a symmetric key block cipher. For Encryption of variable data in blocks of 64- bits, DES first generates a 64-bit key before discarding the last 8 bits to comply with the NSA's prohibition on utilising 56-bit key for DES. The adaptability of DES stems from its ability to function in several definite modes. ECB, CFB, and OFB are few such modes, useful for DES. Using a small and weak key makes it susceptible to a key assault [15], [26].

Data Encryption Three Times (3DES): Since its original release in 1998, the 3DES block cipher has been widely utilised in cryptography. Each data block in 3DES is encrypted, decrypted, then encrypted three times using the DES cipher .Key length can be selected either 112- bit size or of 16 -bits, and the block size/length of 64 bits. With the creation of Triple DES, the programmer can defend against these kinds of assaults without a new block cipher algorithm. Increasing processing power has made cryptanalytic assaults on the original DES cipher possible [29], [30].

In late nineties, J .Daemen, along-with V. Rijmen, developed the symmetric key based, block cipher named as AES. The AES can protect information using, keys of lengths 128, 192, and 256 bits. There are four primary building pieces that make up AES's 128-bit of digital data length. These blocks are handled just like ordered array of bytes, with the state being a 4x4 matrix, on which various operations are performed in rounds. N = 10, 12, &14 rounds are used for full encryption when using keys of lengths 128, 192, and& 256, respectively [31], [32].

B. Schneir, designed a swift algorithm called Blowfish in the last of 20th century considered to be replacement of running encryption algorithms. It's a symmetric key block cipher with a Fiestal structure, having block length of 64-bit and a flexible key length ranging from 32 to 448-bit,[30], [33].

AES surpassed 3DES and DES, & is also faster than other symmetric encryption techniques. Blowfish outperformed other algorithms, according to the data presented[26]. AES beat DES, its modification 3DES, RC2, in terms of requests handled per second and response time under varying user loads.

[34]Author compared AES with RSA based on calculation time, memory usage, and output byte. RSA used the most memory and had the longest encryption time and output byte. Depending on the dynamic textfiles size, and based on the findings, DES spends the least amount of time for encryption and AES uses the least amount of memory, with very little difference in encryption timings.

[30] compared DES, 3DES, and RSA, symmetric and asymmetric key algorithms. Algorithms were analysed for their potential for information protection, encryption time comparison, and measure of throughput. 3DES provides stronger confidentiality and scalability than DES and RSA, making it suitable even though DES uses less memory. DES can be broken by guessing, making it the final secure algorithm. This study compares AES, DES, 3DES, and Blowfish memory building rates, key sizes, CPU utilisation times, and security.

V. MEASURES OF CIPHERS

Throughput:-It is defined as encrypting bits per second, in number & can be defined in terms of Bps, Kbps or higher units i.e. bit rate at which data is encrypted [35]. The design frequency determines throughput. Eq.(1) defines the throughput measure for required hardware & software design , & is the same for both.

To capture multiple elements of performance, synthetic metrics performs blending of multiple non-correlating indications. Using synthetic metric,

Throughput (KBPS) = (Ks * Cc)/Bs…….. (1)

Here:Ks = Code length of key's = No. of round or Cycle Count required; Bs =Block length related to key.

Area/bit: -It calculates the size cost to encrypt a single cipher text as a fraction of the design area to the block size and predicts the fair area needs for comparing ciphers with various block sizes.

Code size (in bytes): -Storage space needed for the cipher code, S-boxes, P-boxes, & flash memory requirements to hold the bits.

RAM capacity (in KB/MB): - The amount of memory required holding intermediate states while the cipher code is being used.

Cycles per block: - Number of encryption cycle /decryption cycle or both cycles per block.

Cycles per byte: - The number of encryption cycles per byte of data / decryption cycle per byte of data.

Energy required: - The amount of energy required to encrypt (or decode, or both) one block.

Energy Efficiency: - Requiring minimal energy and storage space.

Performance efficiency: - It is referred to as the percentage, throughput determined over the space at a constant clock frequency. It is described as estimates of the length in bits and time to process a single cipher text bit.

Efficiency=(Thr)/(As)……(2)

Here (Thr) =Throughput, (As) =Required area

Consider a design with a modest throughput and a tiny area. Increasing resources proportionately to the increase in power is one technique to boost design throughput.

VI. Prerequisites For Systems

The experiment uses AMD A8-6410 with 4GB of RAM. Simulations used the same platform with these specifications, 64-bit processor, operating at 2.80GHz, Windows-10, 64-bit capacity (10.0, Build 18362). The simulation application uses.NET 2013 visual studio's default settings. Computing resources including CPU time, memory, and processing time are heavily utilised by encryption techniques. The various challenges to be resolved while selecting the optimum algorithm for a certain data type and situation include, among others, memory usage, CPU processing time, and cipher execution time. Primary purpose of implementing each of the four algorithms is to examine their strengths, weaknesses, and resource requirements for encryption rate / decryption rate & hence the speed .

Encryption Rate: The ciphering capabilities must be quick enough to fulfil real-time demands, & Encryption rate assess how quickly each case study program encrypts and decrypts information data.

Encryption rate=Tx(end)-Tx (Start)……. (3)

Here Tx(start) is start of encryption process, Tx(end) is end of cryptographic process.

Meantime is difference between starting and end time of encryption taken by algorithm. Size of data & time taken during encryption is proportional. Different Key Size: Since the symmetric technique uses longer variable keys, key management and a sizable portion of the encryption processing are required.

Period of CPU Utilization: System resources including memory, the CPU, and other

components are few requirements of each algorithm, to ascertain, observed period of CPU utilization required by the particular algorithm.

Avalanche effect = [d (o,c)] / [Sf]……. (4)

Where [d (o, c)]: – is a hamming distance between original & encrypted message, [Sf] – is the file size in Megabyte [MB].

The avalanche or diffusion effect is a cryptographic property that gauges an algorithm's security since the output changes significantly when an input is even slightly changed. To demonstrate how well a cryptographic method functions, it is preferable to have a high diffusion rate or high avalanche effect.

Entropy: - Entropy is a metric describing the unpredictable nature of information. High randomization increases the level of bewilderment for the invader, which is necessary. It is possible to determine entropy using Shannon's formula.

Consumption of Memory: -When implemented, different encryption algorithms require different RAM capacities depending on the structural technique, key size, initialization vectors, and number of operations/rounds to be carried out.A minimum amount of memory is preferred because the RAM determines the system's cost.

Simulation Module:-It featured a heterogeneous data module, real-time/static data, a module for testing algorithms, a module for exploring simulation results, and capabilities like dynamic data size of files, encryption parameters, computational speed depending on the number of rounds, associated key length value, encryption ratio, and the use of resources.

VII. OUTCOME & ANALYSIS

The results of running the simulation software with various data inputs are shown, along with the impact of the encryption mode and data payload choices on each approach. For evaluating the response, RSA and symmetric algorithms (DES, 3DES, AES, and Blowfish) have been selected. Encryption speed of the algorithms are not the same. It should be noted that due to its triple phase encryption feature, 3DES always needs more time than DES. Despite having a lengthy (448 bit) key, the Blowfish encryption algorithm beat DES, 3DES, and AES. Blowfish and AES accomplished the encryption more quickly and with less system resources than DES and 3DES, which are known to have security flaws.

For files of the same size, the same parameters for, but with required key size are utilised for each encryption/decryption technique to find the results mentioned in Fig (4) & Fig (5).

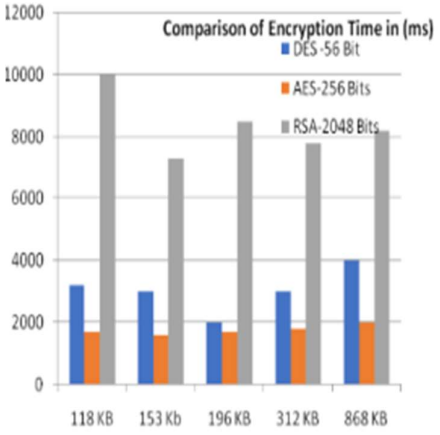


Figure 4 Cipher Tenc

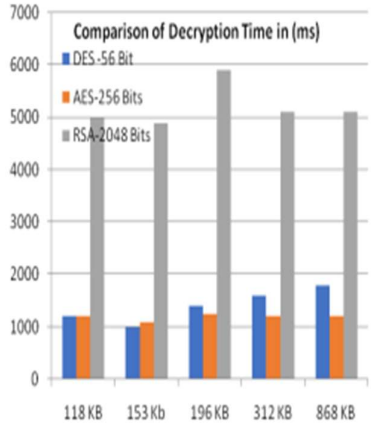
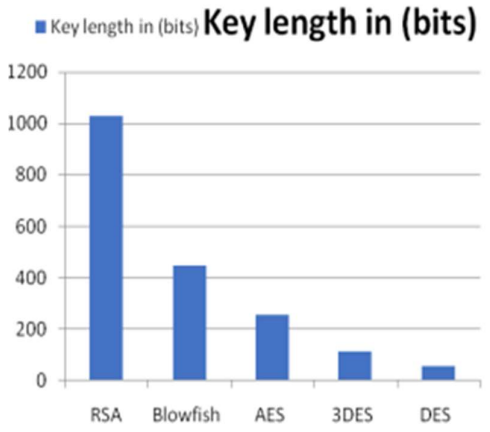


Figure 5 :Cipher Tdec



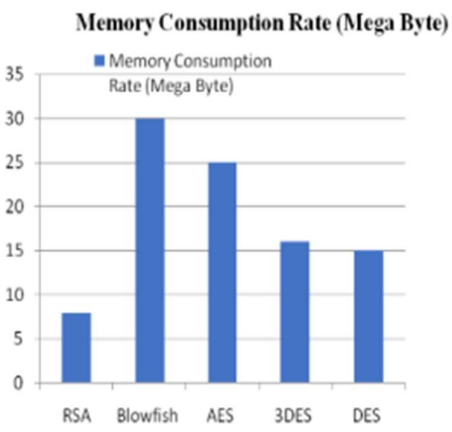


Figure 6: Required Lkey **Figure 7: Memory Rcons**

Four text files of varying sizes are used to compare AES, DES, and RSA encryption/decryption times for corresponding text size.

Comparing the used data text files, and observing simulation response, it is mentioned in Table (1), that AES method consumes less encryption time than RSA. Simulation result observed that, AES cipher module is faster & higher yielding, than DES and RSA. DES utilises less memory than Blowfish. 3DES reuses DES by cascading three instances with different keys. RSA encryption and decoding takes longer than AES and DES. RSA uses the most memory. Blowfish, AES, and 3DES use the same memory for categorization and full encryption. Buffer size varies for each algorithm. RSA algorithm uses the buffer space for different-sized text files. Blowfish is faster than DES. Encryption rate indicates information spread. This proves Blowfish's superior encryption speed.

Figure 6 demonstrates that DES exhibits the least amount of the avalanche effect, whereas Blowfish exhibits the maximum amount. AES employs a substitution permutation over a gliosis field that results in strong information mixing and high output diffusion. It also uses multiplicative inverse and affine transformation.

Overall Throughput = (Ts)/(Te)…….(5)

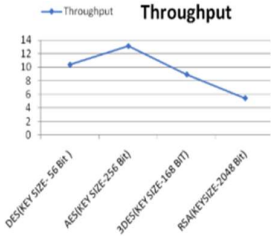
Where (Ts):∑ text size in (MB) in all files,

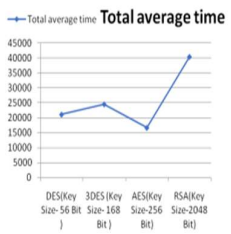
(Te): ∑ evaluation time (ms) for this algorithm

Eq. (5) gives algorithm overall throughput.

AES encrypts more than 13 MB in one second, DES encrypts more than 10 MB, and 3DES encrypts about 8 MB. At 5 MB/Second encryption rate, RSA method performed worse. RSA uses most memory. Blowfish, AES, and 3DES use the same memory for categorization and full encryption.

Fig. 8 shows each algorithm's throughput on the same text files.





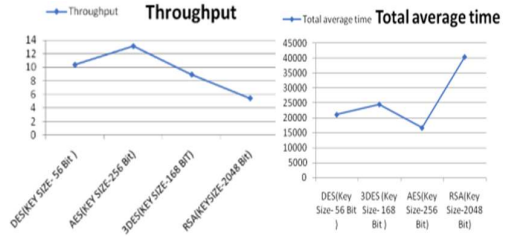


Figure 8: Throughput Comparison Figure 9:Total TAv

VIII. CONCLUDING REMARKS

New communication paradigms between nanodevices, as well as between commonly used micro devices, might be incorporated to accomplish the IoNT goal. This has legitimised using encryption to protect sensitive data. Each encryption method has pros and cons. The best option for ciphering time and necessary memory size is the Blowfish method, which uses more memory, CPU, and time but records the shortest time of all the compared algorithms while using a longer key (448 bits).

Triple phase characteristics of 3DES increase power consumption and reduce throughput. Key length also affects resource utilisation. These findings show that devices with limited memory and power shouldn't use long-key cryptography.

Selection of AES for much better performance in terms of speed along with its prudence and message integrity & can be easily implemented on numerous platforms especially in miniature devices.

If network bandwidth is a concern, DES is optimal. Blowfish and AES can be used to avoid guessing attacks on IPv4 and IPv6-based applications.

Future IoT trends will be built on nanotechnology materials, tiny sensor modules powered by solar power or minimal battery power, sensor nodes and devices, and protocols for the security of sensitive data. The continued expansion of IoT applications will be revolutionised by miniaturisation and nanotechnology combined with IoT protocol.

Table: (1), COMPARISON OF ALGORITHMS BASED ON PARAMETERS

Table: (1), COMPARISON OF ALGORITHMS BASED ON PARAMETERS						
Sr. No.	**Parameters**	**Algorithms**				
		DES.	**3DES.**	**AES.**	**Blowfish.**	**RSA.**
1	**Developed by**	* (IBM)	(IBM)	V. Rijmen, Daeman	Bruce Schneier	Rivest Shamir Adelman
2	**Year of Development**	1977	1998	2001	1993	1978
3	**Characteristics**	Symmetric	Symmetric	Symmetric	Symmetric	Asymmetric
4	**Algorithm Structure**	Feistel	Feistel	SPN	Feistel	Factoring based Modular exponentiation
5	**Block Cipher train**	Binary bits (1/0)	Binary bits (1/0)	Binary bits (1/0)	Binary bits (1/0)	Binary bits (1/0)
6	**Size of Key in Bytes**	07Byte	14Byte, 21Byte	16Byte, 24Byte, 32Byte	04Byte – 56Byte	64Byte – 512 Byte Maintained Key Size >1024 bits.
7	**Rounds for Uniqueness**	16 cycles	48 cycles	(10, 12, 14) cycle for diverse keys	31 cycles	1 cycle
8	**Cipher Block Size Length**	64 Bit Length	64 Bit Length	128 Bit Length	64 Bit Length	Variable, normally =1 Byte, but for(x) bits key size, Block Size= "floor((x-1)/8)" in bytes
9	**Possible Keys**	256	2^{112} 2^{168}	2^{128} ,2^{192} ,2^{256}	2448	RSA key relies on two large prime no. *p* and *q*,

						so infinite output key.
10	**Steps involved in Cipher**	(Expansion & permutation box) +(S-box) +(P-box) +(X-OR)	(Expansion, permutation box) +(S-Box) + (P-box) +(X-R) + Inverse cipher	Substitution (Sub Byte) +Shift-row+ Mix-column + Add round key.	(EX-Or) + (additions on 32-bit)	,
11	**Speed based throughput**	Lower than AES	Lower than DES	Lower than Blowfish	High	Less
12	**Flexibility in extensions**	No	YES, Extended from 56 to 168 bits	YES, 256 key size is multiple of 64	YES, 64-448 key size in multiple of 32	Yes
13	**Security Strengths**	Inadequate security strength	Adequate security strength	Excellent security strength	Excellent security strength	Excellent security strength
14	**Encryption rate [Es]**	Slow rate	Very slow rate	Faster rate	Fast rate	Fast rate
15	**Persuasiveness software (S/W), hardware(H/W)**	Slow response in (S/W) & (H/W)	Slow response in (S/W)	(S /W & H/W) show equal response	(S/W) efficient	(S/W) efficient
16	**Attacks on Ciphers**	Brute –Force, Linear cryptography	Brute -Force attack, recognized plaintext, preferential plaintext	Sidebar (SCA), Biclique attacks	Dictionary attack	Timing attack, Brute Force, Chosen Cipher attack
17	**Memory/Requirements**	2^{43} known plaintexts	2^{32} known plaintexts, 2^{88} memories	.-	Minimum memory size, Maximum 14 round is broken	Highest requirement of memory
18	**Attack complexity**	$2^{39–43}$	:2^{90}	:$2^{126.1}$	.-	.-
19	**Processing Time to check all unique possible keys**	Apx. 1.09 years - 56 bit key- @ 50 billion keys / sec.	Apx. 2.19 years - 112 bit key - @ 50 billion keys / second	Apx. $5*10^{21}$ days- 128 bit keys- @ 50 billion key / sec.	Apx. 8.76 years- @ 50 billion keys / second	.-
20	**Ingest power**	Low Power	Low, but more than DES	Low power	.-	High Power

REFERENCE

[1] S. Mewada, A. Sharivastava, P. Sharma, S. S. Gautam, and N. Purohit, “Performance Analysis of Encryption Algorithm in Cloud Computing,” 2015, [Online].

[2] A. Odeh, S. R.Masadeh, and A. Azzazi, “A Performance Evaluation Of Common Encryption Techniques With Secure Watermark System (Sws),” International Journal of Network Security & Its Applications, vol. 7, no. 3, pp. 31–38, May 2015, doi: 10.5121/ijnsa.2015.7303.

[3] K. K. Patel, S. M. Patel, and P. G. Scholar, “Internet of Things-IOT: Definition, Characteristics, Architecture, Enabling Technologies, Application & Future Challenges,” International Journal of Engineering Science and Computing, 2016, doi: 10.4010/2016.1482.

[4] V. A. Thakor, M. A. Razzaque, and M. R. A. Khandaker, “Lightweight Cryptography Algorithms for Resource-Constrained IoT Devices: A Review, Comparison and Research Opportunities,” IEEE Access, vol. 9, pp. 28177–28193, 2021, doi: 10.1109/ACCESS.2021.3052867.

[5] J. Carlos and S. Fernandes, “Choosing the Future of Lightweight Encryption Algorithms.”

[6] H. Landaluce, L. Arjona, A. Perallos, F. Falcone, I. Angulo, and F. Muralter, “A review of iot sensing applications and challenges using RFID and wireless sensor networks,” Sensors (Switzerland), vol. 20, no. 9. MDPI AG, May 01, 2020.

[7] T. Zhang and M. Livny, “BIRCH: A New Data Clustering Algorithm and Its Applications,” Kluwer Academic Publishers, 1997.

[8] L. Atzori, A. Iera, and G. Morabito, “The Internet of Things: A survey,” Computer Networks, vol. 54, no. 15, pp. 2787–2805, Oct. 2010, doi: 10.1016/j.comnet.2010.05.010.

[9] K. Alsabti and S. Ranka, “CLOUDS: A Decision Tree Classifier for Large Datasets.”

[Online]. Available: http://www.kdnuggets.com/.
[10] O. Goldreich, "Journal of Cryptology © 1993 International Association for Cryptologic Research," 1993.
[11] C. Riman and P. E. Abi-Char, "Comparative Analysis of Block Cipher-Based Encryption Algorithms: A Survey," Computer Fraud, vol. 3, no. 1, pp. 1–7, 2015,
[12] D. Sehrawat and N. S. Gill, "Lightweight Block Ciphers for IoT based applications: A Review," 2018
[13] F. Maqsood, M. Ahmed, M. Mumtaz Ali, and M. Ali Shah, "Cryptography: A Comparative Analysis for Modern Techniques," 2017. [Online]. Available: www.ijacsa.thesai.org
[14] O. Harfoushi and R. Obiedat, "Security in Cloud Computing Using Hash Algorithm: A Neural Cloud Data Security Model," Mod Appl Sci, vol. 12, no. 6, p. 143, May 2018,
[15] Karthik S and Muruganandam A, "Data Encryption and Decryption by Using Triple DES and Performance Analysis of Crypto System." [Online]. Available: www.ijser.in
[16] M. Marjaniet al., "Big IoT Data Analytics: Architecture, Opportunities, and Open Research Challenges," IEEE Access, vol. 5, pp. 5247–5261, 2017,
[17] "A Salad of Block Ciphers The State of the Art in Block Ciphers and their Analysis," 2017.
[18] A. Abidemi Emmanuel, O. E. Aderemi, A. O. Marion, and A. O. Emmanuel, "A Note on Time and Space Complexity of RSA and ElGamal Cryptographic Algorithms." [Online]. Available: www.ijacsa.thesai.org
[19] D. N. Key, S. Singhal, N. Singhal, and M. T. Student, "Encryption algorithm Encrypted Message (Cipher text) Decryption algorithm Plai n Plai n Encryptio n Key A Comparative Analysis of AES and RSA Algorithms," Int J Sci Eng Res, vol. 7, no. 5, 2016, [Online]. Available: http://www.ijser.org
[20] M. Singh, M. A. Rajan, V. L. Shivraj, and P. Balamuralidhar, "Secure MQTT for Internet of Things (IoT)," in Proceedings - 2015 5th International Conference on Communication Systems and Network Technologies, CSNT 2015, Sep. 2015, pp. 746–751. doi: 10.1109/CSNT.2015.16.
[21] R. Masram, V. Shahare, J. Abraham, and R. Moona, "Analysis and Comparison of Symmetric Key Cryptographic Algorithms Based on Various File Features," International Journal of Network Security & Its Applications, vol. 6, no. 4, pp. 43–52, Jul. 2014, doi: 10.5121/ijnsa.2014.6404.
[22] K. S. Mohamed, New Frontiers in Cryptography. Springer International Publishing, 2020. doi: 10.1007/978-3-030-58996-7.
[23] B. J. Mohd, T. Hayajneh, and A. v. Vasilakos, "A survey on lightweight block ciphers for low-resource devices: Comparative study and open issues," Journal of Network and Computer Applications, vol. 58, pp. 73–93, Dec. 2015, doi: 10.1016/j.jnca.2015.09.001.
[24] M. Cazorla, S. Gourgeon, K. Marquet, and M. Minier, "Survey and benchmark of lightweight block ciphers for MSP430 16-bit microcontroller," Security and Communication Networks, vol. 8, no. 18, pp. 3564–3579, Dec. 2015, doi: 10.1002/sec.1281.
[25] N. FasihahMohdEsa, S. Faisal Abdul-Latip, and M. RizuanBaharon, "A Survey of ARX-based Symmetric-key Primitives," 2019.
[26] H. O. Alanazi, B. B. Zaidan, A. A. Zaidan, H. A. Jalab, M. Shabbir, and Y. Al-Nabhani,

"New Comparative Study Between DES, 3DES and AES within Nine Factors," 2010.
[27] B. Seok and C. Lee, "Fast implementations of ARX-based lightweight block ciphers (SPARX, CHAM) on 32-bit processor," Int J Distrib Sens Netw, vol. 15, no. 9, Sep. 2019
[28] K. Zhang, L. Ding, and J. Guan, "Cryptanalysis of Hummingbird-2."
[29] P. Patil, P. Narayankar, D. G. Narayan, and S. M. Meena, "A Comprehensive Evaluation of Cryptographic Algorithms: DES, 3DES, AES, RSA and Blowfish," in Procedia Computer Science, 2016, vol. 78, pp. 617–624.
[30] M. Nazeh, A. Wahid, A. Ali, and M. Marwan, "A Comparison of Cryptographic Algorithms: DES, 3DES, AES, RSA and Blowfish for Guessing Attacks Prevention," 2018, [Online]. Available: www.symbiosisonline.orgwww.symbiosisonlinepublishing.com
[32] B. Padmavathi and S. R. Kumari, "A Survey on Performance Analysis of DES, AES and RSA Algorithm along with LSB Substitution Technique," 2013
[33] P. Patil, P. Narayankar, D. G. Narayan, and S. M. Meena, "A Comprehensive Evaluation of Cryptographic Algorithms: DES, 3DES, AES, RSA and Blowfish," in Procedia Computer Science, 2016, vol. 78, pp. 617–624.
[34] O. G. Abood and S. K. Guirguis, "A Survey on Cryptography Algorithms," International Journal of Scientific and Research Publications (IJSRP), vol. 8, no. 7, Jul. 2018
[35] A. Pratap, R. Gupta, V. S. S. Nadendla, and S. K. Das, "Bandwidth-constrained task throughput maximization in IoT-enabled 5G networks," Pervasive Mob Comput, vol. 69, Nov. 2020, doi: 10.1016/j.pmcj.2020.101281.
[36] M. Javaid, A. Haleem, R. P. Singh, S. Rab, and R. Suman, "Exploring the potential of nanosensors: A brief overview," Sensors International, vol. 2. KeAi Communications Co., Jan. 01, 2021. doi: 10.1016/j.sintl.2021.100130.
[37] C. L. Grigsby, Y. P. Ho, and K. W. Leong, "Understanding nonviral nucleic acid delivery with quantum dot-FRET nanosensors," Nanomedicine, vol. 7, no. 4. pp. 565–577, Apr. 2012. doi: 10.2217/nnm.12.28.
[38] C. A. E. Hauser, S. Maurer-Stroh, and I. C. Martins, "Amyloid-based nanosensors and nanodevices," Chemical Society Reviews, vol. 43, no. 15. Royal Society of Chemistry, pp. 5326–5345, Aug. 07, 2014. doi: 10.1039/c4cs00082j.
[39] W. Di and H. A. Clark, "Optical nanosensors for: In vivo physiological chloride detection for monitoring cystic fibrosis treatment," Analytical Methods, vol. 12, no. 11, pp. 1441–1448, Mar. 2020, doi: 10.1039/c9ay02717c.
[40] J. M. Dubach, D. I. Harjes, and H. A. Clark, "Fluorescent ion-selective nanosensors for intracellular analysis with improved lifetime and size," Nano Lett, vol. 7, no. 6, pp. 1827–1831, Jun. 2007, doi: 10.1021/nl0707860.
[41] L. Fang et al., "Turning bulk materials into 0D, 1D and 2D metallic nanomaterials by selective aqueous corrosion," Chemical Communications, vol. 55, no. 70, pp. 10476–10479, 2019, doi: 10.1039/c9cc04807c.
[42] N. Tang, C. Zhou, L. Xu, Y. Jiang, H. Qu, and X. Duan, "A Fully Integrated Wireless Flexible Ammonia Sensor Fabricated by Soft Nano-Lithography," ACS Sens, vol. 4, no. 3, pp. 726–732, Mar. 2019, doi: 10.1021/acssensors.8b01690.
[43] I. F. Akyildiz, J. M. Jornet, and C. Han, "Terahertz band: Next frontier for wireless communications," Physical Communication, vol. 12, pp. 16–32, 2014,
[44] W. Zhang, S. Zhu, R. Luque, S. Han, L. Hu, and G. Xu, "Recent development of carbon

electrode materials and their bioanalytical and environmental applications," Chemical Society Reviews, vol. 45, no. 3. Royal Society of Chemistry, pp. 715–752, Feb. 07, 2016. doi: 10.1039/c5cs00297d.

[45] K. Kiani, "Magnetically affected single-walled carbon nanotubes as nanosensors," Mech Res Commun, vol. 60, pp. 33–39, 2014, doi: 10.1016/j.mechrescom.2014.05.005.

[46] H. H. Pajooh, M. Rashid, F. Alam, "Multi-layer blockchain-based security architecture for internet of things," Sensors (Switzerland), vol. 21, no. 3, pp. 1–26, Feb. 2021,

[47] R. Abdel-Karim, Y. Reda, and A. Abdel-Fattah, "Review—Nanostructured Materials-Based Nanosensors," J Electrochem Soc, vol. 167, no. 3, p. 037554, Jan. 2020, doi: 10.1149/1945-7111/ab67aa.

[48] S. Aleksic, "A survey on optical technologies for IoT, smart industry, and smart infrastructures," Journal of Sensor and Actuator Networks, vol. 8, no. 3. MDPI AG, 2019. doi: 10.3390/jsan8030047

[49] .Dume, I., (2018) Spray-on antennas for the Internet of Things. [online] Physics World. Available at: https://physicsworld.com/a/spray-on-antennas-for-the-internet-of-things/