iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** IX **Month of publication:** September 2022

DOI: https://doi.org/10.22214/ijraset.2022.46697

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue IX Sep 2022- Available at www.ijraset.com

Identifying and analyzing Risk Mitigation strategies in IOT devices using Light Weight Symmetric Encryption Algorithms

Saurav Verma[1], Mahek Pokharna[2], Vishal Mishra[3]
[1, 2, 3]*Department of I.T Mukesh Patel School of Technology and Management, NMIMS (Deemed-to-be) University, Mumbai, India*

Abstract: ***Internet of Things (IoT) is becoming more and more pervasive in all applications. It has greater capabilities like remote monitoring and control. Different available APIs make IoT devices and applications easy to develop and deploy. The data generated by IoT devices is smaller in size and needs light weight protocols like MQTT to carry it over the network. Risk Mitigation in IoT data is very crucial and to do that traditional security and risk mitigation algorithms like RSA, SHA-512 and all cannot be used as IoT devices have smaller data. Applying traditional security and Risk Mitigation techniques to IoT data by traditional algorithms will cause the computation overhead in IoT applications. Different Light Weight Encryptions schemes for risk mitigation are suggested like PRESENT, HUMMINGBIRD et al. in this paper, different light weight encryption algorithms used in IoT risk mitigation are studied and understood. Their problems are noted down and possible improvements are suggested to make them more efficient.***

Keywords: Risk Mitigation, Risk Analysis, Light Weight Encryption, Light Weight Encryption Algorithms, IoT risk mitigation, Comparing Light weight Encryption Algorithms, problems and solutions.

I. INTRODUCTION

Internet of Things provides a mechanism to transfer the data over a network of interconnected systems without the human-computer interaction or the human-human interaction. The entities in an IOT system can be people, animals or a system of connected mechanical or computer devices wherein each entity has its own distinct identifier and can communicate internally. IOT is of utmost importance for businesses for reducing the costs by automation of mundane processes. The sensor data obtained from all the objects interconnected in a system is shared and used for analytics purposes to extract the information from the raw data and utilize them for fulfilling the business needs. The sensor data is collected from the interconnected IOT devices and is sent to a IOT gateway which is then sent to the cloud for further analysis of the data. The interconnected devices can also communicate with each other for the data required for their individual functioning with least human intervention. IoT has found its applications in many fields like:

A. Home

The owners of the network can collect the sensor information from the network with the help of Wi-Fi to enable data transfer of larger bandwidth (videos) along with sampling rates in the higher end of the spectrum. The home appliances if connected with the Internet of Things, will provide an improved sense of management for energy control. Social networking can have a huge impact on the applications of IoT as well [1]. An interesting concept like 'IOTagram' can be used where each appliance connected via a network can post its reading on Instagram with the connection of a single network and keep other devices updated. [2-3]

B. Agriculture

The soil being the most essential component of farming can be used to monitor the agricultural patterns and their results with the help of IoT. The sensors can be utilised to capture data like the moisture level of the soil, the proportion of the different chemicals, current temperature and hence gain insights on whether the soil is suitable for farming by checking the state of the soil. IoT would prove to be conducive to the farmers in aspects like irrigation control and water management for efficient farming.

C. Healthcare

The fit-bands or wearables can be used to stay connected with patients and get their health information continuously monitored to identify any abnormalities and be proactive.

This would help in taking augmented care of the patients and would limit the rate of fatalities. A concept of smart beds can be used wherein the IoT can be integrated with the hospital beds to get the metrics like oxygen level or temperature being remotely monitored.

II. PROBLEM STATEMENT

Privacy of data and its security is of top priority in IOT devices as huge chunks of personalized data is involved in IoT applications. Applying traditional security and Risk Mitigation techniques to IoT data by traditional algorithms will cause the computation overhead in IoT applications hence there is a need of study and review of modern and lightweight risk mitigation algorithms for IOT devices.

III. RISK MITIGATION IN IOT

Risk Mitigation of IOT devices is of top priority as huge chunks of data is involved in IoT applications. The networks handling the data in IoT applications need to be robust and should be able to withstand the security attacks. The actual data of IoT applications is stored in the cloud storage which is a third party service and the user would require the extra assurance of the privacy and security of their confidential data. This can be achieved by adhering to compliances and completing the certifications with the help of cloud audits [11,17].

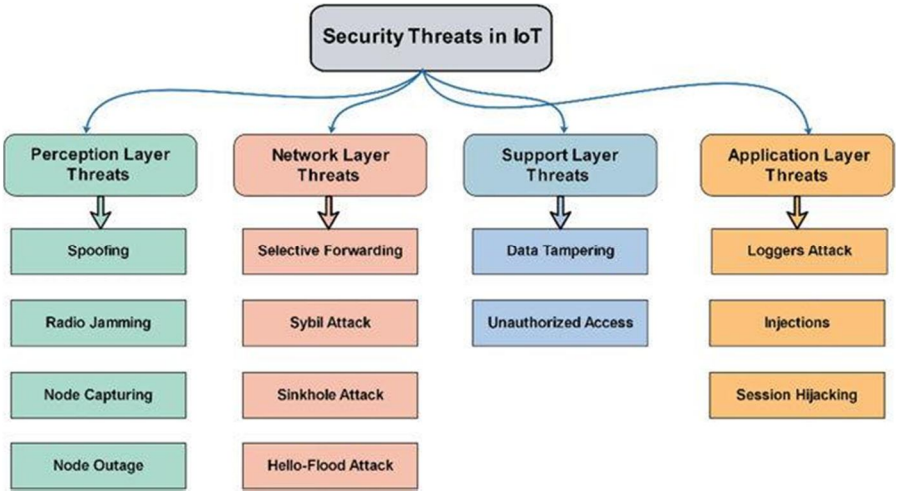


Fig. 1. The IoT Security Layers

A. *Perceptual Layer*

The perceptual layer majorly comprises the devices which are responsible for capturing the data from the environment like the Camera, RFID tags and the sensors. These devices can easily be attacked as they are openly placed in the environment. Some of the major risks can be as follows -

1) *Fake Node:* An attacker can simply place a new node in the network and inject the virus into the network which would in turn bring the entire system down [12].
2) *Physical Damage:* The devices in the perceptual layer being physically placed in the environment are more susceptible to physical damages as the devices like cameras and sensors can be easily damaged and hence a DoS attack can be invoked.
3) *Virus Injection:* The attacker can inject the virus into a node that he may have unauthorized access to, and can inject the malicious code into the actual node and gain access to the confidential use data [13].

B. *Network Layer Security*

Even after implementing security algorithms for risk management, the network layer has certain issues to be addressed. These risks can affect the privacy and coherence of the data.

1) *Congestion:* The humongous data being fetched by the myriad number of devices in the network and the communication of such huge amounts of data amongst the devices leads to the congestion issues in the network [12].
2) *Noise Interference:* In this attack the attackers tend to induce the noise in the Radio frequency signals and hence disrupt the actual communication by manipulating the RFID signals [11].

C. Middleware Layer Security

1) *Data Security:* The data obtained from the IoT sensors is stored in cloud for analytics purpose. This also adds up to the data security measures needed to be taken by means of database monitoring to detect the unauthorized data migration or data loss prevention activities.
2) *Virtualization Security:* The virtualization techniques used by the varying cloud vendors overlook some of the compliances and security measures as they adopt different mechanisms for virtualizations [14].

D. Application Layer Security

1) *Access and Authorization:* The different users in the system may have different access rights and so the security to check the levels of access of data of the users must be implemented [12].
2) *Phishing Attacks:* The use of malicious links provided in emails from falsified senders pretending to be actuals users to steal the data from the genuine user [16].
3) *Application Layer Security Requirements:* The users must be provided the knowledge on using the combinations of characters to create a password which is strong enough and difficult for the attacker to guess [15].

E. Traditional Risk Mitigation Approaches

Different traditional risk mitigation approaches are also introduced here, to provide the better insight of the security algorithms.

1) *Cryptographic Techniques:* The cryptographic techniques which comprise the symmetric key algorithms like Advance encryption standard (AES) and asymmetric algorithms like Rivest Shamir Adelman (RSA) cannot be used for the security of the IoT applications [10]. The higher CPU configurations required by these algorithms makes them expensive for their implementation in this area. Hence a new need of creating more feasible security algorithms has come up. The symmetric key algorithms use the same key for encryption and decryption at the sender as well as the receiver's end. On the other hand the asymmetric algorithm uses different keys for encryption and decryption. The sender encrypts the data to be sent on the network using the public key and sends it in the form of cipher text and the receiver decrypts this cipher text using his own private key. The asymmetric algorithms require appropriate key management as multiple keys are involved and hence become complex to implement as compared to symmetric key algorithms where only a single key is involved.
2) *Key Management:* The most vital point of implementation of any cryptographic algorithm is key management. In the IoT architecture, the raft number of connected nodes in an IoT network and the myriad amount of data hinders the existing key management techniques [6-8].
3) *Denial of Service:* The Denial of Service (DoS) attack launches a large number of requests to the system and makes the system unavailable for serving actual requests. The applications in IoT in the healthcare domain can lead to loss of lives in case a DoS attack is launched on them. The battery operated IoT devices prevent the DoS detection on the sensor nodes as even sample attack messages cannot be launched on the IoT devices to detect and occlude them [5].
4) *Authentication and Access Control:* It is feasible to implement Authentication of devices before communication of data between a limited numbers of devices with protocols like SSL handshake. As far as IoT is concerned a large number of devices are involved in data communication amongst themselves and so authentication of data using these existing protocols is not feasible. Thus more research has to be done to fulfill the authentication and access control of the nodes within an IoT network.

These traditional security approaches are not suitable to secure the data in IoT due to different time and space complexities of IoT context. The solution to this problem is we need Light Weight encryption techniques.

IV. RISK MITIGATION USING LIGHTWEIGHT ENCRYPTION

The main focus of lightweight encryption is to optimize the cryptographic algorithms based on standard cryptographic primitives to run on small and resource constrained devices. The aim is to provide the authentication and encryption in one pass by ensuring the communicating entities that their information is not tampered with. The IoT devices require less-intensive computational resources and lower power consumption due to the utilization of battery.

The features of lightweight encryption are -

A. Speed

The set of instructions can execute faster and hence provide the results at a much faster rate. This would benefit in getting the insights into the data quickly as the raw data is captured by the sensors.

B. Power Consumption

The execution of the set of instructions takes place faster and hence the system can return into an idle mode as fast as possible to minimize the power utilization. This helps the IoT devices to function in a more efficient manner as they are battery operated and need minimum power utilization.

C. Computation

The Light weight encryption is supposed to run on the IoT devices which handles smaller data but greater in numbers. So this encryption scheme should take minimum computation power, as IoT devices have limited computing capacity.

V. LITERATURE REVIEW

Light weight encryption scheme is more suitable for the devices like RFID (Radio Frequency Communication) and WSN (Wireless Sensor Networks) etc. In light weight encryption schemes the data should be of small sizes with small keys. The techniques to do encryption and decryption should be very much less in numbers to cause lighter computations. Also the memory used by such operations is very less. Light weight encryption schemes use logical operations like XOR, AND, OR and NOT [18]. They are also divided as stream ciphers and block ciphers. They are also designed to produce the hash values to check for data integrity. But these functions and hashes are also light weight processes. Also one can use different programming utilities like left shift and right shift operators to implement faster permutation and combinations of bits. The symmetric encryption schemes take lesser execution time than the asymmetric encryption schemes [19]. So in this paper, different symmetric key encryption algorithms are considered. In this paper, different light weight symmetric key encryptions algorithms for Risk Mitigation like PRESENT, HUMMINGBIRD, DESL, AES, HIGHT and TWINE are considered. Each algorithm is studied in details and the comparison is shown in table 1.

A. Present

PRESENT [20] is a lightweight symmetric key block encryption algorithm. It consists of 31 normal rounds and one final round which is the mixing step. The block is of 64 bit and the key can be of 80 bit or 128 bit and it has 64 bit plain and cipher text. The single S-box, which is of 4 bit, acts as the basis of the nonlinear layer, which is parallelly applied 16 times per round and was designed keeping in mind hardware optimizations. It involves bit-oriented permutation and is based on the Substitution Permutation Network [21]. Its implementation requirements are similar to compact stream ciphers. It has three different architectures- Round-based, Pipelining, and Serialized. PRESENT is around 2.5 times smaller as compared to AES. It also has low power consumption and is applied in situations where high chip efficiency is required.

B. Hummingbird

Hummingbird algorithm is a light-weight algorithm used for encryption in IOT devices like wireless sensors and radio frequency identification. It consists of a 128-bit encryption secret key and an initialization vector of 64-bit. This algorithm combines property of both, stream and block cipher and mirrors the Helix and Phelix proposal. Hummingbird algorithm encrypts and decrypts any payload, with associated data, such as the nonce and the header of packet, using the Authenticated Encryption with Associated Data method [25]. Security against cryptanalysis attacks and other common cyber-attacks is provided by this algorithm, since it provides a small block size. It is adroit in environments with resource constraints and it can carry out large virtual rtos with custom block ciphers. [26]

C. DES Light (DESL)

The DESL is a lightweight cryptographic algorithm, which is used for lightweight applications, like in passive RFIDs and other IOT based sensors. The DESL algorithm is similar to DES and based on the traditional DES, with slight modifications. [28] The DESL repeatedly uses single S-box eight times. The data encryption standard does not perform well in constrained environments, unlike the lightweight DES algorithm. The DESL provides security against cyber-attacks like linear cryptanalysis. The goal is to minimize probability of collision, in the S-box output, while implementing the DESL. [29]

D. AES

Advanced Encryption Standard (AES) [31] is a symmetric and iterative encryption algorithm. It is a modified version of the Rijndael block cipher [32]. It uses a fixed block length. It has 128-bit data and key sizes of 128 (10 rounds), 192 (12 rounds) or 256 (14 rounds) bit.

The 128 bit internal state is set to the block for plaintext initially, and then it becomes the output block containing cipher text, after transformations. It performs computations on bytes. Thus the 128 bits of the plaintext block are treated as 16 byte which are converted into a four row and four column matrix. Substitution–permutation network forms the base of AES. It is made of operations involving replacement of inputs by specific outputs, called substitution, and shuffling around of bits, called permutations. Both these operations are linked together serially. It is faster as compared to Triple-DES and requires less power. It also provides high-security as it is implemented in both software and hardware.

E. Hight

Hight Algorithm is a new light weight encryption block cipher with 64-bit block length and 128-bit key length. The Hight Algorithm refers to low-resource hardware implementation, which is proper to computing device such as a sensor in USN or a RFID tag. Hight is a secure algorithm used in various cryptographic applications. It is implemented where there is a requirement for less cost, less use of power, and ultra-light implementation. It consists of simple operations such as XOR, addition mod and left bitwise rotation. Hight algorithm is a variant of the generalized Feistel network and has a 32 round iterative structure. It is more of hardware oriented rather than software oriented. [36]

F. Twine

Twine Algorithm is a block cipher algorithm which presents a 64-bit lightweight block cipher. It requires a relatively small amount of hardware implementations and it enables efficient software implementations on various platforms, from micro-controller to high-end CPU. Twine makes use of an extremely efficient nonlinear layer using 4-bit S-boxes and a diffusion layer, which manages the 16 blocks. Twine allows a compact implementation of unified encryption and decryption. For security, it employs a specific technique to improve the low diffusion rate of GFS, however, it is the primacy to evaluate the security against attacks. It is a variant of the Type- 2 GFS. [37]

All above listed algorithms are compared for their different aspects like specific application, to the specific platforms they run and the comments are highlighting the strengths.

Table 1. Comparison Light Weight Encryption Algorithms for Risk Mitigation

Sr. No.	Algorithm Name	Application	Platform	Comments
1	PRESENT [43]	RFID	Hardware	Have used minimal data path, round based data path and minimal data path
2	HUMMINGBIRD [44]	RFID tags	Hardware	Works for active and passive tags
3	DESL [45]	RFID	Hardware	Can prevent multiple DES vulnerable attacks
4	AES [42]	Not given	software	128-bit key length is successfully implemented on 3 platforms
5	HIGHT [46]	RFID using FPGA	Hardware	Implemented in scalar and pipelined mode
6	TWINE [47]	Not Given	Hardware	Saturation and diff. crypt analysis is not possible

VI. PROBLEMS FACED IN DIFFERENT ALGORITHMS

Different light weight encryption algorithms shown in table 1, are studied and the problems encountered in their applications, deployments and complexities are listed down. These problems are related to cipher text round, key generation, key sharing to IoT device deployment issues.

A. Present

The architecture of PRESENT makes it susceptible to some dedicated forms of attacks like attack using palindromic differences and some advanced variants of differential linear attacks [22].

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue IX Sep 2022- Available at www.ijraset.com

Moreover there are no established guidelines to the design of key schedules which can lead to a wide variety of schedule-specific attacks like related-key attacks [23] and slide attacks [24]. Both of these rely on building easily identifiable relationships between different sets of sub keys.

B. Hummingbird

A major problem with Hummingbird algorithm is the tradeoff among security, cost, and performance. Finding an optimal cost-performance ratio metrics is an impediment to this encryption algorithm. The throughput of the hummingbird algorithm without a pipelining approach is less. Similarly, the efficiency of encryption and decryption without pipelining is less. The modulo addition is slower since it has high number of logic elements, and thus the time cost for encryption and decryption is higher.

C. DES light (DESL)

The key length of the DES algorithm is not sufficient to use in today's times and it might prove to be obsolete. The DESL algorithm needs to provide substantial security by adding extra features and elements to avoid attacks like linear cryptanalysis. There may be a tradeoff between the cost and trust on the algorithm. [30]

D. AES

AES is secure against brute force and mathematical attack but it is susceptible to timing attacks. Timing attacks are implementation level attacks which mean they depend on the input. These attacks are possible as AES implementations perform S-box lookups which depend on the key and take variable time. Cache-collision timing attacks have been proven to work against AES [33-34].

E. Hight

High Energy Consumption: The single round implementation of the HIGHT encryption requires that the hardware iterates 32 cycles, one cycle for each transformation round. As such, the consumed energy to encrypt a single block is noticeably high which is a huge problem as more and more power is required to iterate through each cycle. Low Efficiency: The scalar version of the Hight algorithm is not capable of multiprocessing which leads to low efficiency and throughput and affects the speed of the processes. Evaluation: The proposed architectures are coded in Verilog. They are programmed codes are synthesized, and the area is measured using Synopsys Design Compiler. This affects the combinational logic of the system. [38]

F. Twine

Poor Diffusion: The drawback of Twine is poor diffusion property due to its design resulting in a small but-slow cipher due to many rounds. Low throughput: The design leads to low throughput and power but can be improved and enhanced through a solution. Low Speed: Sometimes multiprocessing leads to low speeds which can be enhanced by using advanced design structures. [39]

VII. SOLUTIONS

Different solutions to the problems faced in light weight encryption algorithms for risk mitigation are found and suggested.

A. Present

Though the attacks work promisingly over a few rounds, they start losing their practical value and are unlikely to harm the PRESENT cipher [20]. And to counter key schedule based attacks, a counter, that is dependent on rounds, is used so that sub-key sets cannot be "slid" easily. Non-linear operations enable the efficient mixing of the contents of the key register.

B. Hummingbird

A pipelining approach is indicated to solve the problem and increase the throughput and efficiency of the hummingbird algorithm. Replacing the obsolete modulo addition with the XOR operation reduces the number of logic elements, enabling a faster processing of data and data packets at the time of encryption and decryption.

Reduction of the instruction count through pipelining ensures optimized processing of the data during encryption/decryption process. This algorithm, with the pipelining concept, can attain a substantially higher efficiency and throughput as compared to AES, SEA, with a smaller area requirement.

Thus, it is an ideal lightweight cryptographic algorithm. [27]

C. DES Light (DESL)

The key whitening idea introduced in DESX avoids the problem of the insufficient key length in DES. An improved S-box can be selected to provide a higher output, by reducing the collisions in the S-box. Another solution to increase efficiency is to XOR the values of bits at several positions, rather than dealing with one-bit position. Designing the S-box conditions in the correct way strengthens the algorithm efficiency and makes it stronger and more resistant to attacks like linear and differential cryptanalysis. [29]

D. AES

Timing attacks can be avoided by selecting primitives that facilitate efficient constant-time implementations or by avoiding the use of S-box [35]. But this type of selection is difficult and would result in slowing down the process. Adding delays to the comparatively faster operations can also help in hiding the timing differences.

E. Hight

The best approach to reduce the energy of the design is to implement multiple transformation rounds in the hardware as to reduce the number of iterations spent on the transformation round and hence reduce the energy dissipated by the flip-flops. The Hight algorithms consists of a pipelined design which is capable of processing more than one task at a time and thus is expected to have better throughput. The round-based implementation added need to be compared only with serial implementations. [40]

F. Twine

We can substantially improve the diffusion property of Type-2 GFS by using a different block shuffle from the original cyclic shift. TWINE is efficient on software and enables compact unification of encryption and decryption. TWINE uses neither a bit permutation nor a Galois-Field matrix. In the speed-first implementation, two rounds are processed in one loop. This leads to removal of the block shuffle between the first and second rounds to enhance the performance. A further speeding up is possible if more rounds are contained in one loop at the cost of increased memory. [41]

VIII. PERFORMANCE COMPARISON OF RISK MITIGATION ALGORTIHMS

Encryption algorithm uses different factors like key size, number of rounds, block size, s-Box and operations like shifting, transformations, confusion and diffusion of bits. Every parameter and every operation needs computation resources and time. The table 2, shows the overall summary of parameters.

Sr. No.	Algorithm	Key Size (bits)	Total Rounds	Block size (bits)	Energy need	Using S-Box	Orientation Hardware or Software
1	Present	80, 128	31	64	low	Yes	hardware
2	Humming Bird	128	4	16	Low	Yes	Both
3	Desl	128	64	64	High	Yes	Both
4	AES	128	10	128	High	Yes	Both
5	Hight	64	32	64	Low	No	hardware
6	Twine	80, 128	43	64	moderate	yes	hardware

From the table 2, many conclusions can be drawn. The IoT devices should be very much energy efficient because in certain places like oil wells, country borders, heating chambers, it is not possible to replace the battery. And again from the time and the complexity view the AES and DESL algorithms are higher. Since, the DESL and AES algorithms are not energy efficient, uses more number of rounds and the key length is also more in bits. So in the current context they are not highly suitable. The Twine algorithm is moderate in terms of energy and also used 43 rounds in the process. So in terms of overall efficiency it is better than AES and DESL. The Next algorithm is Hight which is low in terms of energy and uses 32 number of rounds, but it is only hardware oriented plus do not have trusted s-Box security. So Hight is better than twine. Present algorithm is taking 31 rounds with moderate key sizes, it's less in energy need and only hardware oriented. So present comes before the Hight in terms of efficiency.

Hummingbird algorithm takes only 4 rounds, it is hardware and software oriented, and it uses s-Box and simple logical operations for cipher generation. It can generate output of 16-bits. And it is less in energy and can be implemented on hardware and software both. Plus this algorithm is also applicable for Arduino type of micro-controller devices where output is in 10 bit resolution. Thus, hummingbird is the best algorithm out of all covered algorithms.

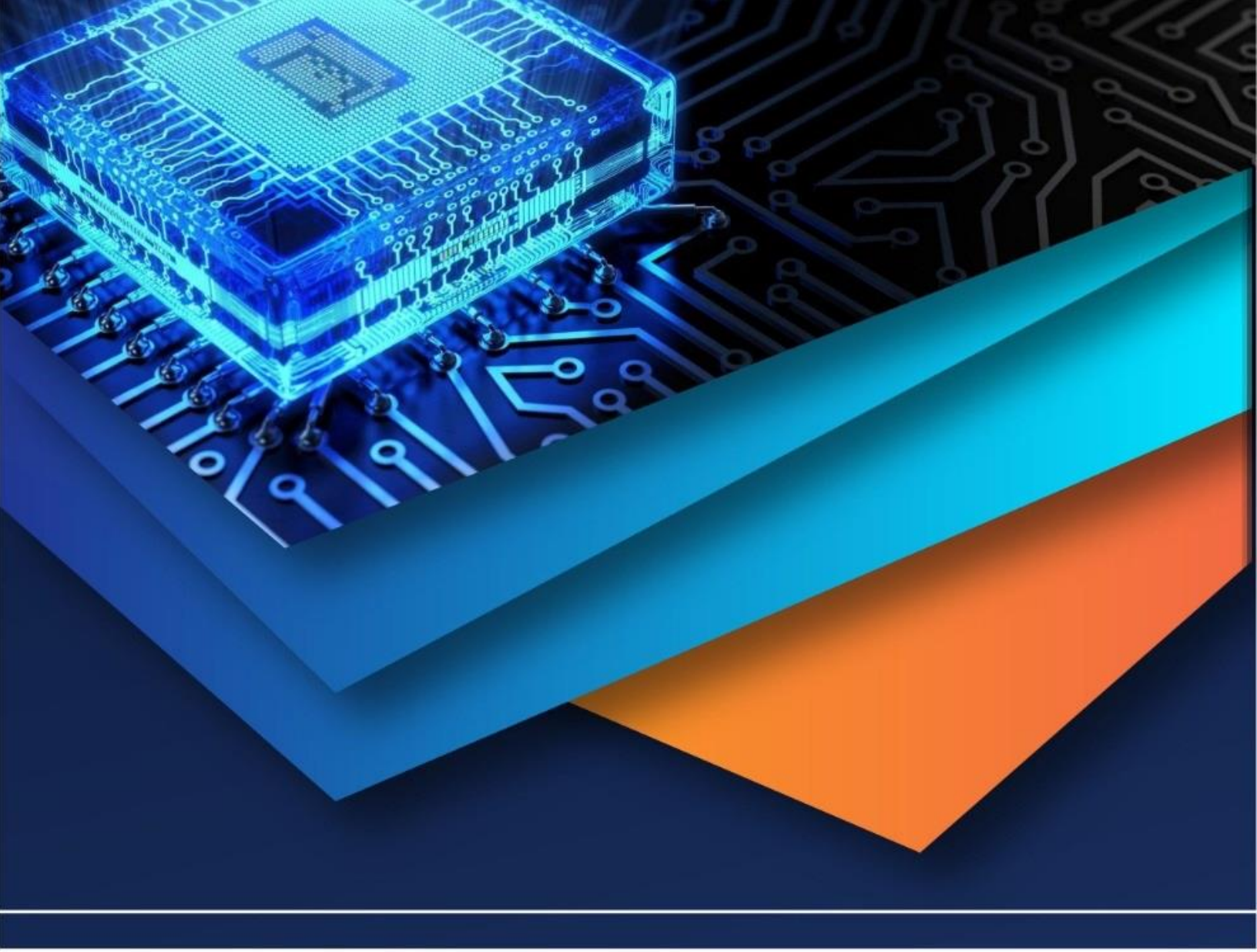
IX. CONCLUSION AND FUTURE WORK

Main idea behind the Internet of Things with its applications and need for risk mitigation is discussed. Also, there is the need of newer and lightweight security algorithms to mitigate the risks which have come forth due to the complexities in the implementation of existing risk mitigation algorithms in IoT applications. Light Weight Encryption Algorithms are existing for stream ciphers, block ciphers. They are also using the mechanism of either symmetric key encryption or asymmetric key encryption. To provide data integrity different light weight hash algorithms like PHOTON etc. are also present. Different IoT applications have different security needs for risk mitigation. Light Weight Encryption Algorithm Schemes are able to cover all these security needs and makes the IoT applications more secured and less computation intensive. In this paper different light weight risk mitigation encryption algorithms are studied and compared. One great aspect of these algorithms is that they can be deploy on either hardware or software form. If software application is not able to do the computation then to provide the security, one can use hardware which can be embedded to the IoT data source. The hardware chips can do the encryption/decryption very fast, but the problem is they add up to the hardware and makes more bulky in terms of energy computation, handiness and mobility. In future, different light weight algorithms like TEA and LEA can be studied and analyzed in terms of risk mitigation. The above problem solutions can be incorporated into the protocols to make them more secured and again they are subjected for the future analysis and the need.

REFERENCES

[1] H.S. Ning, Z.O. Wang, Future Internet of Things architecture: like mankind neural system or social organization framework? IEEE Communications Letters 15 (2017) 461–463.
[2] Nikoloski, K. The role of information technology in the business sector. Int. J. Sci. Res. (IJSR) 2019, 3, 303–309.
[3] X. Li, R.X. Lu, X.H. Liang, X.M. Shen, J.M. Chen, X.D. Lin, Smart community: an Internet of Things application, IEEE Communications Magazine 49 (2020) 68–75.
[4] Rochmah, T.N.; Fakhruzzaman, M.N.; Yustiawan, T. Hospital staff acceptance toward management information systems in Indonesia. Health Policy Technol. 2020, 9, 268–270.K.Sonar, H.Upadhyay, "A survey DDoS attack on Internet of Things.", International journal of Engineering research and Development. Volume 1-, issue 11 (November 2014)
[5] S. Shantharajah, K. Duraiswamy, G. Nawaz, "Key Management and distribution for authenticating group communication.", First international on Industrial and Information System, ISSN: 2164-7011, IEEE
[6] Mutwiri, W. Amazon Business Information Systems. Data Acquisition and Management in Its Value Chain; GRIN Verlag: Mu-nich/Ravensburg, Germany, 2020.
[7] W. Abdullah, N. Boudriga, D. Kim, S. An, "An efficient and Scalable key management mechanism for Wireless Sensor Networks.", 16th International Conference on Advanced Communication Technology. ISSN: 1738-9445, IEEE
[8] Ahmad, S.; Alam, K.M.R.; Rahman, H.; Tamura, S. A comparison between symmetric and asymmetric key encryption algorithm based decryption mixnets. In Proceedings of the 2015 International Conference on Networking Systems and Security (NSysS), Dhaka, Bangladesh, 5–7 January 2015; pp. 1–5.
[9] Yassein, M.B.; Aljawarneh, S.; Qawasmeh, E.; Mardini, W.; Khamayseh, Y. Comprehensive study of symmetric key and asymmetric key encryption algorithms. In Proceedings of the 2017 International Conference on Engineering and Technology (ICET), Antalya, Turkey, 21–23 August 2017; pp. 1–7
[10] Lucas, H.C., Jr. Performance and the use of an information system. Manag. Sci. 2020, 21, 908–919.
[11] K. Zhao, L. Ge, "A Survey on the Internet of Things Security." Computational Intelligence and Security, Ninth International Conference, IEEE 2021
[12] Abe, S.; Ozawa, M.; Kawata, Y. Science of Societal Safety: Living at Times of Risks and Disasters; Springer Nature: Berlin/Heidelberg,Germany, 2019.
[13] The treacherous 12, Cloud Computing top threats 2016, "Top threats working group, Cloud Security Alliance (CSA)"
[14] C. Ding, L. J. Yang, and M. Wu, "Security architecture and key technologies for IoT/CPS", ZTE Technology Journal, vol. 17, no. 1, Feb.
[15] T. N. Jagatic, N. A. Johnson, M. Jakobsson, and F. Menczer, "Social phishing." Communications of the ACM 50, no. 10 (2007): 94-100.
[16] S. Ali,S. Sabbir, Z. Ullah, "Internet of Things Security, Device Authentication and Access Control: A Review", Ninth International Conference, IEEE 2018
[17] Juntunen, M.; Lehenkari, M. A narrative literature review process for an academic business research thesis. Stud. High. Educ.2021, 46, 330–342.
[18] Exploring the Differences Between Symmetric and Asymmetric Encryption, https://cyware.com/news/exploring-the-differences-between-symmetric-and-asymmetric-encryption-8de86e8a
[19] Bogdanov, A., Leander, G., Knudsen, L.R., Paar, C., Poschmann, A., Robshaw, M.J., Seurin, Y. and Vikkelsoe, C., "PRESENT - An Ultra-Lightweight Block Cipher", Cryptographic Hardware and Embedded Systems (CHES), Lecture Notes in Computer Science, vol. 4727, 2017, pp. 450-466, DOI:10.1007/978-3-540-74735-2_31, Springer.
[20] Aven, T.; Zio, E. Foundational issues in risk assessment and risk management. Risk Anal. 2018, 34, 1164–1172.
[21] M.E. Hellman and S.K. Langford, "Differential-Linear Cryptanalysis", Proceedings of Crypto, LNCS, vol. 839, 1994, pp. 17–25, DOI:10.1007/3-540-48658-5_3, Springer-Verlag.
[22] E. Biham, "New Types of Cryptanalytic Attacks Using Related Keys", Proceedings of Eurocrypt, LNCS, vol. 765, 1994, pp. 398–409, DOI:10.1007/BF00203965, Springer-Verlag.

[23] Aven, T. Risk assessment and risk management: Review of recent advances on their foundation. Eur. J. Oper. Res. 2020, 253, 1–13.
[24] Engels D., Saarinen MJ.O., Schweitzer P., Smith E.M. (2020) The Hummingbird-2 Lightweight Authenticated Encryption Algorithm. In: Juels A., Paar C. (eds) RFID. Security and Privacy. RFIDSec 2011. Lecture Notes in Computer Science, vol 7055. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-25286-0_2
[25] Engels, Daniel & Fan, Xinxin & Gong, Guang & Hu, Honggang & Smith, Eric. (2019). Hummingbird: Ultra-Lightweight Cryptography for Resource-Constrained Devices. Financ. Cryptogr. Data Secur.. 6054. 3-18. 10.1007/978-3-642-14992-4_2.
[26] Pranali Umap, Dr. A.S.Joshi (2017). Performance Analysis of Hummingbird Cryptographic Algorithm using FPGA. Journal of Multidisciplinary Engineering Science and Technology (JMEST). ISSN: 3159-0040. Vol. 2 Issue 11
[27] Zodpe, Harshali & Wani, Prakash & Mehta, Rakesh. (2012). Design and implementation of algorithm for DES cryptanalysis. 278-282. 10.1109/HIS.2012.6421347.
[28] Aboshosha, Bassam & Dessouky, Mohamed & El-Sayed, Ayman. (2019). Immunity of Lightweight DES Algorithm (DESL) Against Linear Cryptanalysis Attack.
[29] A. Poschmann, G. Leander, K. Schramm and C. Paar, "New Light-Weight Crypto Algorithms for RFID," 2017 IEEE International Symposium on Circuits and Systems, New Orleans, LA, 2007, pp. 1843-1846, doi: 10.1109/ISCAS.2007.378273.
[30] National Institute of Standards and Technology (NIST), Advanced Encryption Standard (AES), 2001, http://www.csrc.nist.gov/publications/fips/fips197/fips-197.pdf, FIPS-197.
[31] Daemen, J. and Rijmen, V., ” The Design of Rijndael: AES — The Advanced Encryption Standard”, 2002, DOI:10.1007/11836810_13, Springer-Verlag.
[32] Bonneau J. and Mironov I., ”Cache-Collision Timing Attacks Against AES”, Cryptographic Hardware and Embedded Systems (CHES), Lecture Notes in Computer Science, vol. 4249, 2016, pp. 201-215 DOI:10.1007/11894063_16, Springer.
[33] D. J. Bernstein, “Cache-timing attacks on AES”, 2018, http://cr.yp.to/antiforgery/cachetiming-20050414.pdf.
[34] Abdullah Al Hasib and Abul Ahsan Md. Mahmudul Haque, “A Comparative Study of the Performance and Security Issues of AES and RSA Cryptography”, International Conference on Convergence and Hybrid Information Technology, vol. 2, 2018, pp. 505-510, DOI: 10.1109/ICCIT.2008.179, IEEE.
[35] Hong D. et al. (2006) HIGHT: A New Block Cipher Suitable for Low-Resource Device. In: Goubin L., Matsui M. (eds) Cryptographic Hardware and Embedded Systems - CHES 2016. CHES 2016. Lecture Notes in Computer Science, vol 4249. Springer, Berlin, Heidelberg.
[36] Kobayashi, Eita & Suzaki, Tomoyasu & Minematsu, Kazuhiko & Morioka, Sumio. (2017). TWINE: A Lightweight Block Cipher for Multiple Platforms. Selected Areas in Cryptography. 7707. 10.1007/978-3-642-35999-6_22.
[37] Y.-I. Lim, J.-H. Lee, Y. You, and K.-R. Cho, “Implementation of HIGHT cryptic circuit for RFID tag,” IEICE Electronics Express, vol. 6, no. 4, pp. 180–186, 2009.
[38] Canni`ere, C.D., Dunkelman, O., Knezevic, M.: KATAN and KTANTAN - A Family of Small and Efficient Hardware-Oriented Block Ciphers. In: Clavier and Gaj [15], pp. 272–288.
[39] N. Mouha, B. Mennink, A. Van Herrewege, D. Watanabe, B. Preneel, and I. Verbauwhede, “Chaskey: an efficient MAC algorithm for 32-bit microcontrollers,” in Selected Areas in Cryptography–SAC, pp. 306–323, Springer, Berlin, Germany, 2016.
[40] Daemen, J., Knudsen, L.R., Rijmen, V.: The Block Cipher Square. In: Biham, E. (ed.) FSE. Lecture Notes in Computer Science, vol. 1267, pp. 149–165. Springer (2016).
[41] Bos, J.W., Osvik, D.A., Stefan, D.: Fast implementations of AES on various platforms. IACR .(2016)
[42] Rolfes,C., Poschmann, A., Leander, G., Paar,C.: Ultra-Lightweight Implementations for Smart Devices—Security for 1000 Gate Equivalents. Springer, Germany (2020).
[43] Fan, X., Hu, H., Gong, G., Smith, E.M., Engels, D.: Lightweight Implementation of Hummingbird Cryptographic Algorithm on 4-Bit Microcontrollers. IEEE (2021).
[44] Ghafari, V.A., Hu, H., Chen, Y.: Fruit-v2: ultra-lightweight stream cipher with shorter internal state. Int. Assoc. Cryptol. Res (IACR) (2016).
[45] Jungk, B., Lima, L.R., Hiller, M.: A Systematic Study of Lightweight Hash Functions on FPGAs. IEEE (2018).
[46] Aumasson, J.-P., Henzen, L., Meierm, W., Naya-Plasencia, M.: Quark: a lightweight hash. CHES (2018).



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY