Research Article

IoT Storage Optimisation Security Framework Based on a Secure Hash Algorithm

Aditiya Agnihotri

Lecturer, Department of CSE (Computer sc)

GEHU-Dehradun Campus

Abstract: The purpose of this research is to examine the risks that IoT applications encounter throughout operational operations, data routing, and energy utilisation, as well as potential solutions to these problems utilising Blockchain technology. This study is broken up into four parts: finding a secure hash technique for the huge volumes of data produced by IoT devices; improving storage; using cluster-based routing to keep out unwanted visitors; and safeguarding IoT programmes. The trust of the blockchain node to existing applications for the Internet of Things cannot be assured, and a sizable amount of communication assets would be used in the process of achieving consensus using the proposed security evolution method of Hybrid Cryptographic Hash Function (HCHF). Private Consensus Bitcoin with Hybrid Cryptographic Hash Function (HCHF) Enhanced Trust on Multilink was a newly developed consensus method in this study. The following are some of the key features of the IoT applications that make use of the suggested HCHF method: 1) The confidence of the nodes in the IoT application grew as a result of the development of the trust attribute mechanism, leading to the formation of a grouping structure within the network. The trust in the system is boosted by the grouping adjustment mechanism. Second, the complexity of communication is considerably simplified by categorising nodes according to their level of trust. The experimental findings show that the system's communication effectiveness and overall trust may be greatly increased by the blockchain's application of the upgraded method.

Keywords: Hybrid Cryptographic Hash Function (HCHF), IoT applications, Private Consensus Blockchain, cluster-based routing

Introduction

The Internet of Things (IoT) is a new era made possible by the development of wireless networking, the Network's underlying infrastructure, and distributed computing. Various communication methods are used to establish connections between these systems and the Internet. Communication between these modules within an IoT ecosystem is dependent on several factors, such as storage optimisation, configuration, standards, security concerns, and a wide range of services associated to different possible ways.

To extend the bounds of the earth with virtualized physical things, the present IoT development, also known as the network of the future, comprises of billions of differently connected items or devices that make use of current technology. The IoT has had a major effect on the rapidly evolving sector of the economy, and its influence is only expected to grow in the years to come. It is predicted that by the end of 2020, over 50 billion items and machines will be connected to the Internet, fostering further development and innovation in IoT [1]. With the Internet as its backbone, IoT enables sensors and other devices to analyse data, carry out predetermined activities, store information in the cloud, and dynamically set off alarms in the event of a catastrophic event. Therefore, the Internet of Things makes use of a wide range of technological advances, including ubiquitous computing, machine learning (ML), wearable systems, a number of different networking protocols and systems, a wide range of network equipment, and a number of different Web protocols, to improve the performance of traditionally used products.

By interconnecting various items and devices, the IoT hopes to provide better services. The massive amount of traffic that will be created in the next decades by IoT nodes, which are estimated to number in the billions, is driving the growth of storage optimisation and security challenges in IoT. Therefore, the Internet of Things requires an infrastructure built to reduce the burden of this vast data on other systems that rely on wireless and other types of networks. If system vulnerabilities are not fixed, the lack of security will also hinder future IoT expansion. In order to help readers, this study optimises their storage space and improves the safety of their Internet of Things (IoT) applications by considering a wide variety of indications and cutting-edge technologies. This will pave the way for further investigations into the issue. Systems like smart buildings, tomorrow's supermarkets, smart cities, smart transportation monitoring, and many more rely on the information received by the Internet of Things.

This data will be sent to the end-user or customer upon request or on a constructive level. All 0's, followed by 60's, 80's, 100's, and 120' Percentage of Total Searches Annual Search Patterns on Google 2020-2021 Predictions for Google Searches The Worldwide Web of Things Distributed Ledger Technology (Blockchain): IoT safety: (all around the world) Optimisation of Storage Facilities: (Global) information to the user, however, might cause a wide range of technical issues. Most peers in a distributed system's transport routing protocol are sleeping to save energy; those who are closer to the sink module, however, must be roused so that data may be collected and sent to the base station without interruption or wasted energy. Selecting the optimal solution from a set of alternatives using a linear regression model may greatly increase the useful lifespan of a system. In addition to the aforementioned factors, services like personal monitoring systems and other medical items need the dependability and safety of packet transmission when sending data through an IoT system. Since IoT is a part of internet architecture, which covers many different industries and applications, these concerns must be addressed if transmission power is to be used and managed properly for IoT. IoT expansion might be stunted if problems with networks like these aren't fixed.

Secured Hash Algorithm

Blockchain Technology's ability to register all interactions between IoT devices on a decentralised public ledger guarantees data security and authenticity without relying on a central authority. With the exception of the Internet of Things (IoT), which relies on centralised computers known as mainframes, the Blockchain has no single point of failure [2]. The decentralised nature of the Internet of Things relies on blockchain technology, and the design of this technology helps to reduce the likelihood of security breaches. However, there are a few significant problems that might be addressed by developing solutions for block chain in IoT [3]. Since 1) mining is computationally costly due to the architecture of resource-constrained Sensor networks, and 2) block processing is time-consuming for many other time-based important applications, and 3) the IoT network is only expected to expand, a more scalable blockchain network is required [4]. Among all these extensive storage of interactions or transfers in the database backup system, verifying only limited interactions and ensuring speedy response while keeping privacy is a tough problem [5]. Therefore, under the blockchain storage architecture, not only are huge data processing and optimisation non-trivial, but so is data security. To guarantee that no two sets of transfer records are ever combined in the same way, the blockchain, as a decentralised database, organises transaction data into an ascending series of blocks, each of which is secured by encryption technology. New blocks in the blockchain are only added once the PoW consensus algorithm has performed competitively. The new hash value, the hash of the preceding block, and the contents are all kept in a block [6]. The system mandates a distributed consensus process that regulates (i) the incorporation of new units into the block chain framework, (ii) the reading mechanism for secure prevent chain confirmation, and (iii) the integrity of the received data of transaction information contained in each version of the blockchain preserved on every block [7].

A Peer-to-Peer network is an example of a distributed computing paradigm if each participant in the network has access to all of the network's processing, storage, and sensing resources. All the information on the structure and operation of the network can only be provided by these widely used services. Since it is decentralised, Blockchain does not yet have the infrastructure to support data analysis tools or user orders. Therefore, a node stands in for a process actor[33]. This decentralisation and uniform service provision across all platforms is made possible since any participant may act as a server and a client. Since the blockchain is decentralised, there is no need to worry about a single point of failure, and data may be concentrated to make decoding it more difficult. However, this criteria is inadequate to guarantee the security and uniformity of data throughout the process. Blockchain uses verification to achieve this goal. There are many different kinds of cryptographic methods, and the data bits they provide vary depending on the hashing technique used. The blockchain design often makes use of private and public keys as encryption primitives for cryptographic hash methods. The public key is needed to verify the sender's identity and the transaction's origin, making it necessary for any physically encrypted transactions. It encrypts with one key and decrypts with another[8]. It is practically difficult to track out only the encryption parameters that were utilised to produce two distinct codes. This safeguards the integrity and legitimacy of the findings if anybody recognises their transaction using the secret key, since decryption relies heavily on the universally-origin hash function.

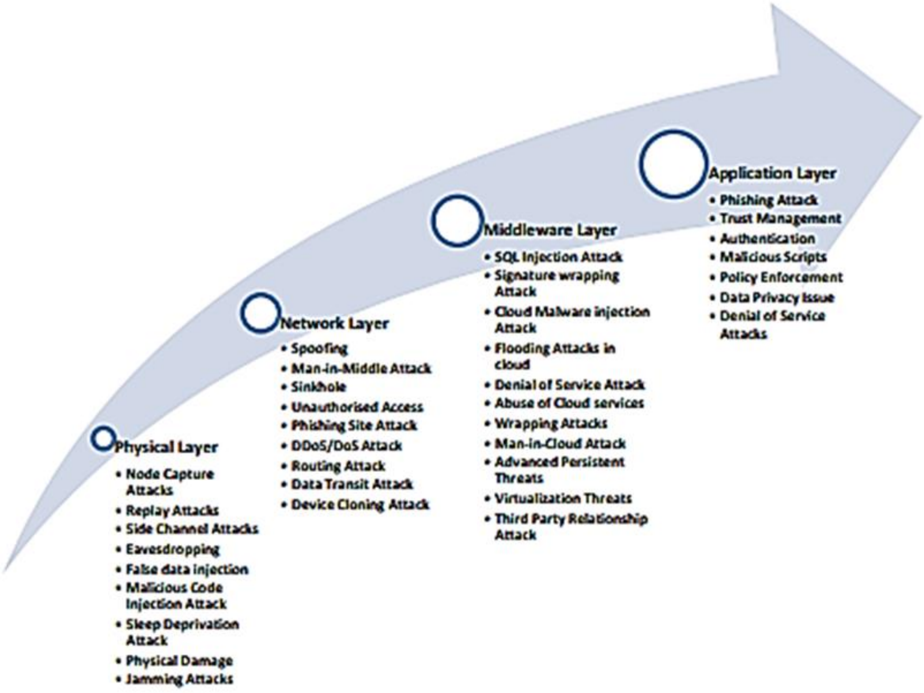

Figure 1. Security Threats at IoT Architectural Levels

Problems with security in various IoT applications, network congestion, insufficient load bearing capacity of Cloud Servers and services, implications with device architecture, and data manipulation all arise as IoT grows rapidly.

Literature Review

S. Huh et al.[1] For scenarios with millions of connected IoT devices, the limitations and difficulties of the current server-client approach have been projected. Therefore, we advocate for the use of blockchain technology in the creation of an IoT framework. They used blockchain for tracking and personalising IoT gadgets. To manage keys, they relied on RSA public key cryptosystems, with the public keys stored in Ethereum and the secret keys stored on individual nodes. The writers opted for Ethereum as their shared database due to the fact that its consensus process would allow them to easily create apps that would operate atop it. This allows them to efficiently manage IoT setup settings and build up a crucial infrastructure. Ethereum was chosen because it provides a means of efficient system control, something that is lacking in most other blockchain frameworks despite the widespread acceptance of transactions as a crucial foundation. Instead of using a whole IoT framework with thousands of IoT systems for the proof of concept, they employ only a few to demonstrate the notion. Later, they said that they had successfully employed blockchain to build a fully scalable IoT platform.

H. Suo et al.[2] IoT applications continue to provide considerable challenges, and it has been stated that data security is a major issue. The authors have conducted IoT research and given specific attention to security in order to make this new field more approachable. A comprehensive analysis of the security system and its features is used to establish the security requirements. Based on this, they discussed how far we've come in studying dangers and addressing long-term trends like authentication, network bandwidth, sensor data security, and cryptography.

S. Sicari et al.[3] IoT security and privacy was covered, including security of data and encryption, protection mechanisms inside the Internet of Things network, client and issue safety and confidence, and the execution of security regulations. Conventional security preventive measures cannot be explicitly implemented to IoT applications due to the varying needs and networking frames contained. A scalable infrastructure that can handle possible threats in a fairly complicated setting is necessary, and this is made more difficult by the enormous number of connected devices. The authors discussed the most pressing issues and solutions in IoT security research, as well as future questions and directions for study.

L. Atzori et al. [4] argued that the key facilitator of this promising new paradigm is the confluence of several breakthroughs and security systems. Decentralised information for intelligent devices, enhanced communication standards (thanks to the next Generation Internet), and monitoring and verification are among the most significant breakthroughs in recent years. It seems to reason that any really significant contribution to the development of the Internet of Things would be the result of collaborative efforts spanning several information disciplines, including but not limited to networking, information systems, electronics, and the social sciences. Given the intricacies of the problem, this research is meant for people interested in learning more about this area and making contributions to its development.

In this study, we gather information using six separate sensor units to measure things like air quality index, voltage level, temperature, humidity, and water level. After everything is processed, a comparison is made between several Secured hash algorithms. Then, a hash algorithm is chosen after a thorough three-tiered assessment of secured hash algorithms. After settling on a hash, the SHA256 instance will be used to compress the data before it is uploaded to the cloud, as seen in Figure 2.

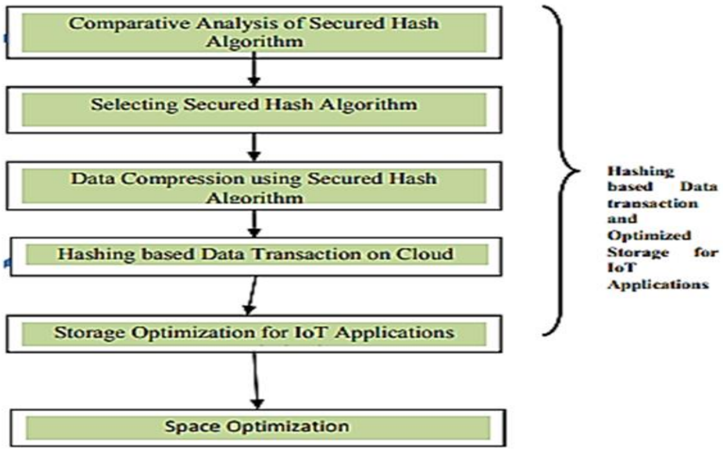
Comparative Analysis of Secured Hash Algorithm
Selecting Secured Hash Algorithm
Data Compression using Secured Hash Algorithm
Hashing based Data Transaction on Cloud
Storage Optimization for IoT Applications
Space Optimization
Hashing based Data transaction and Optimized Storage for IoT Applications

Figure 2: Phase I- Framework for Hashing based Data transaction and Optimised Storage for IoT Applications

Evaluation of Secured Hash Algorithms and Storage Optimization

Different types of cryptographic algorithms provide a unique set of data bits in their hashes. The blockchain design often makes use of private and public keys as encryption primitives for cryptographic hash methods. The public key is needed to verify the sender's identity and the transaction's origin, making it necessary for any physically encrypted transactions. To encrypt and decode, it makes use of separate keys. It is practically difficult to track out only the encryption parameters that were utilised to produce two distinct codes. This safeguards the findings' integrity and validity if anybody recognises their transaction using the secret key, since decryption relies heavily on the universally-originative hash function. If the decryption is successful, the client is certain that the concealed key came from the person who validated the contract and that the data has not been tampered with or altered. These operations encode information of many kinds into a string of characters. This identical understanding will return virtually a definite key, but any variation in the underlying data will produce a unique code. The process of creating a hash is quite simple and requires little computing power, however the inverse is not true. It is impossible to modify the procedure and retrieve the original data even if the hash value is known. After the new structure has been built, it is checked for validity using cryptographic techniques. If someone attempts to tamper with the chain of blocks, the cypher text will change and the blockchain will reflect the new information. The outcome is the presence of cons in such areas. If an attacker tampers with a blockchain block and its associated key, the block's value will no longer correspond to the sequence number of the subsequent chain of blocks.

Implementing Cryptographic Hash Algorithm

There are several possible implementations of the secure hash algorithms. Use of FPGA, Xilinx, Java, C, etc. are all examples. The algorithm it employs is universally consistent between implementations. The following procedures may be used to implement SHA-256: By adding 64 bits, the input is made exactly 64 bits smaller than a multiple of 512. One must be the initial value in the combination, with zeros used to fill in any gaps. Adding 64 more bits to the information makes it a multiple of 512. You may calculate those 64 bits of characters by calculating the modulus of the unpadded real data. Third, initial values for eight rounds' worth of buffers must be established. In addition, the array must be able to store 64 unique keys, with values ranging from K [0] to K [63]. 4. The data is broken up into several 512-bit chunks Each block undergoes 64 rounds of calculation, with the results providing input to the following stage. 5. The output of the frame is sent into the following phase as an input at each iteration. If the result is fewer than 512 bits, the procedure starts again; otherwise, it is used as the final hash digest. This hash will be 256 bits in length, as the SHA-256 method implies. Likewise, the following are the required steps for implementing MD5: After receiving the data, step one is to check if the length of the source string is 512 bits less than 64 bits. To add the extra characters with the superfluous bits, one must first enter a 1. Two more characters are needed at the end of the string to make it a multiple of 512. To do the same, you may just calculate the size of the incoming data and divide it by 64. Combining these two procedures makes the final sequence hashable. The whole string is broken down into 512-bit chunks. A, B, C, and D are the four separate buffers that must be initialised. The standard size for buffers is 32 bits. Third, each 512-bit block is divided into 16 32-bit chunks. Each of the four process cycles makes full use of all of the available sub-blocks, buffers, and a predetermined amount of array elements.

T[1] -> T[64] is the name of array.

M[0] -> M[15] denotes sub-blocks.

The buffers B, C, and D will go into non linear process that include the processes as Iteration 1: (b AND c) OR ((NOT b) AND (d)) Iteration

2: (b AND d) OR (c AND (NOT d)) Iteration

3: b XOR c XOR d Iteration

4: c XOR (b OR (NOT d))

Comparative Analysis of Secured Hash Algorithm

During the process, three types of correlation are carried out based on factors like the number of bits in the cryptographic hash's output, the size of the document, the time needed to implement the document using a hashing algorithm, and the influence of the hash function on the speed with which the document is processed. Secured hash algorithms, message digest hash algorithms, and race integrity primitives evaluation message digests are only few of the hash algorithms utilised in blockchain. Figure 3 displays a further categorization of the aforementioned algorithms. Different hash algorithms are better suited to different uses, so the first step in the proposed research is to conduct a comparative analysis of hash algorithms based on three criteria: output size in bits, file size and execution time, and speed performance.

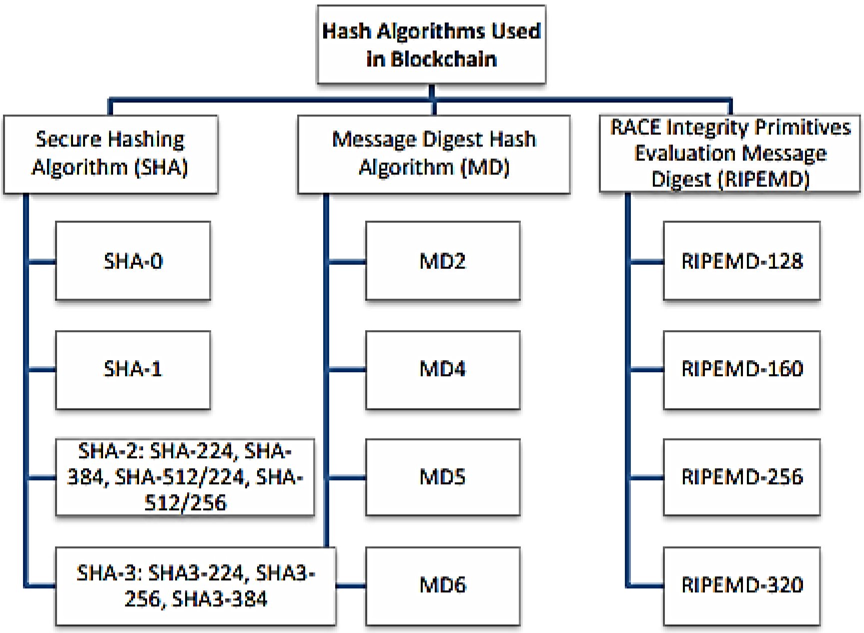
Hash Algorithms Used in Blockchain
Secure Hashing Algorithm (SHA)
Message Digest Hash Algorithm (MD)
RACE Integrity Primitives Evaluation Message Digest (RIPEMD)
SHA-0
SHA-1
SHA-2: SHA-224, SHA-384, SHA-512/224, SHA-512/256
SHA-3: SHA3-224, SHA3-256, SHA3-384
MD2
MD4
MD5
MD6
RIPEMD-128
RIPEMD-160
RIPEMD-256
RIPEMD-320

Figure 3: Types of Hash Algorithms used in IoT

Comparative Analysis on the basis of Output Size

The size of the hash's output (in bits) is used as a metric to compare various hash algorithms. Table 1 displays the output size for various hash functions, based on a variety of hash algorithms and their modifications.

Table 1 Output Size (bits) for Different Hash Algorithms

Algorithm and variant		Output size (bits)
MD5		128
SHA-0		160
SHA-1		
SHA-2	SHA-224 SHA-256	224 256
	SHA-384 SHA-512	384 512
	SHA-512/224 SHA-512/256	224 256
SHA-3	SHA3-224 SHA3-256 SHA3-384 SHA3-512	224 256 384 512

As a result, it can be concluded that the performance bits of SHA-512 and SHA-256 are equivalent as depicted in figure 4

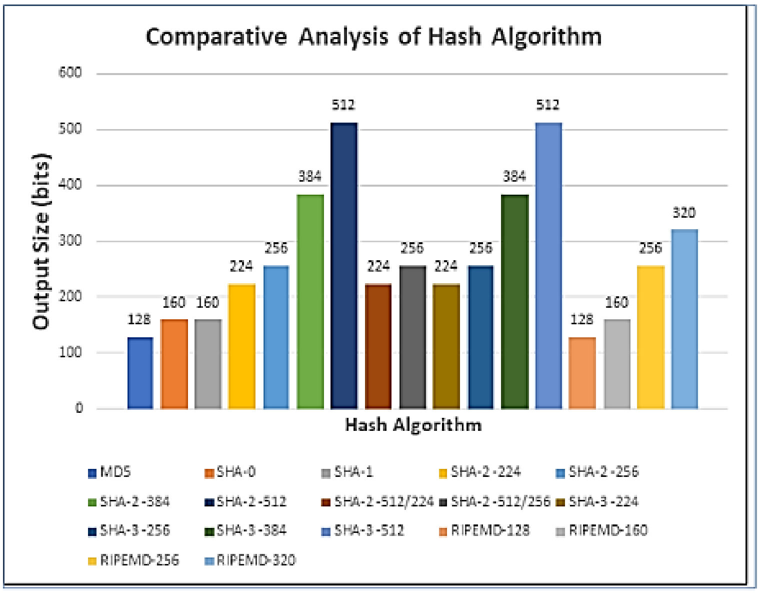

Figure 4. Comparative analysis of Hash Algorithm based on Output Size (bits)

Conclusion

In this paper, we explore the issue of data security and privacy for IoT devices and provide a solution. The blockchain is connected with IoT and used to increase security and enhance the efficiency with which IoT applications are stored. The Internet of Things (IoT) relies on sensors to construct its smart environment, and further sensors will be developed in the future for usage in areas such as smart healthcare and smart supply chains. The many security threats to each architectural layer of IoT infrastructure were the subject of this chapter. By proving an error deviation rate from the original data of less than 1%, we were able to optimise storage by compressing it by 50% using AES (in this case, SHA-256). Further, it is emphasised that the highest space optimisation is 92.2% when 3800 sensor readings are considered, despite the fact that the compressed data is always 11.8 KB for the various file sizes.

References

1. S. Huh, S. Cho, and S. Kim, ―Managing IoT devices using blockchain platform,‖ International Conference on Advanced Communication Technology, ICACT, pp. 464–467, 2017, doi: 10.23919/ICACT.2017.7890132
2. F. A. Alaba, M. Othman, I. A. T. Hashem, and F. Alotaibi, ―Internet of Things security: A survey,‖ Journal of Network and Computer Applications, vol. 88, no. March, pp. 10–28, 2017, doi: 10.1016/j.jnca.2017.04.002
3. S. Sicari, A. Rizzardi, L. A. Grieco, and A. Coen-Porisini, ―Security, privacy and trust in Internet of things: The road ahead,‖ Computer Networks, vol. 76, pp. 146–164, 2015, doi: 10.1016/j.comnet.2014.11.008
4. L. Atzori, A. Iera, and G. Morabito, ―The Internet of Things: A survey,‖ Computer Networks, vol. 54, no. 15, pp. 2787–2805, 2010, doi: 10.1016/j.comnet.2010.05.010.
5. T. Ahram, A. Sargolzaei, S. Sargolzaei, J. Daniels, and B. Amaba, ―Blockchain technology innovations,‖ 2017 IEEE Technology and Engineering Management Society Conference, TEMSCON 2017, no. 2016, pp. 137–141, 2017, doi: 10.1109/TEMSCON.2017.7998367
6. M. Conoscenti, A. Vetro, and J. C. De Martin, ―Blockchain for the Internet of Things: A systematic literature review,‖ Proceedings of IEEE/ACS International Conference on Computer Systems and Applications, AICCSA, vol. 2016, doi: 10.1109/AICCSA.2016.7945805
7. A. Ouaddah, A. Abou Elkalam, and A. Ait Ouahman, ―FairAccess: a new Blockchain-based access control framework for the Internet of Things,‖ Security and Communication Networks, vol. 9, no. 18, pp. 5943–5964, 2016, doi: 10.1002/sec.1748
8. K. Biswas and A. B. Technology, ―Securing Smart Cities Using Blockchain Technology,‖ 2016 IEEE 18th International Conference on High Performance Computing and Communications; IEEE 14th International Conference on Smart City; IEEE 2nd International Conference on Data Science and Systems (HPCC/SmartCity/DSS), pp. 1392–1393, 2016, doi: 10.1109/HPCC-SmartCity-DSS.2016.0198
9. H. A. A. Al-Kashoash, Y. Al-Nidawi, and A. H. Kemp, ―Congestion-aware RPL for 6L0WPAN networks,‖ Wireless Telecommunications Symposium, vol. 2016-May, 2016, doi: 10.1109/WTS.2016.7482026
10. E. Cerritos, F. J. Lin and D. Bastida, "High scalability for cloud-based IoT/M2M systems," 2016 IEEE International Conference on Communications (ICC), 2016, pp. 1-6, doi: 10.1109/ICC.2016.7511050
11. J. Sun, J. Yan, and K. Z. K. Zhang, ―Blockchain-based sharing services: What blockchain technology can contribute to smart cities,‖ Financial Innovation, vol. 2, no. 1, 2016, doi: 10.1186/s40854-016-0040-y
12. T. Ahram, A. Sargolzaei, S. Sargolzaei, J. Daniels, and B. Amaba, ―Blockchain technology innovations,‖ 2017 IEEE Technology and Engineering Management Society Conference, TEMSCON 2017, no. 2016
13. P. Wang, R. Valerdi, S. Zhou, and L. Li, ―Introduction: Advances in IoT research and applications,‖ Information Systems Frontiers, vol. 17, no. 2, pp. 239–241, 2015, doi: 10.1007/s10796-015-9549-2

14. O. Flauzac, C. Gonzalez, and F. Nolot, ―New security architecture for IoT network,‖ Procedia Computer Science, vol. 52, no. 1, pp. 1028–1033, 2015, doi: 10.1016/j.procs.2015.05.099
15. S. Sicari, A. Rizzardi, L. A. Grieco, and A. Coen-Porisini, ―Security, privacy and trust in Internet of things: The road ahead,‖ Computer Networks, vol. 76, pp. 146–164, 2015, doi: 10.1016/j.comnet.2014.11.008
16. S. Das, A. Dey, A. Pal, and N. Roy, ―Applications of Artificial Intelligence in Machine Learning: Review and Prospect,‖ International Journal of Computer Applications, vol. 115, no. 9, pp. 31–41, 2015, doi: 10.5120/20182-2402
17. A. Betzler, C. Gomez, I. Demirkol, and J. Paradells, ―CoCoA+: An advanced congestion control mechanism for CoAP,‖ Ad Hoc Networks, vol. 33, pp. 126–139, 2015, doi: 10.1016/j.adhoc.2015.04.007
18. J. Jermyn, R. P. Jover, I. Murynets, M. Istomin and S. Stolfo, "Scalability of Machine-to-Machine systems and the Internet of Things on LTE mobile networks," 2015 IEEE 16th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2015, pp. 1-9, doi: 10.1109/WoWMoM.2015.7158142
19. A. Bader and M. -S. Alouini, "Blind Cooperative Routing for Scalable and Energy-Efficient Internet of Things," 2015 IEEE Globecom Workshops (GC Wkshps), 2015, pp. 1-6, doi: 10.1109/GLOCOMW.2015.7414088.
20. T. M. Silva Filho, B. A. Pimentel, R. M. C. R. Souza, and A. L. I. Oliveira, ―Hybrid methods for fuzzy clustering based on fuzzy c-means and improved particle swarm optimization,‖ Expert Systems with Applications, vol. 42, no. 17–18, pp. 6315–6328, 2015, doi: 10.1016/j.eswa.2015.04.032