Research Article

Data Authentication for Web of Things (WoT) by Using Modified Secure Hash Algorithm-3 (SHA-3) and Salsa20 Algorithm

Hayder Najm[1] , Rehab Hassan[2], Haider K. Hoomod[3]

[1]Department of Computer Science, University of Technology
[2]Computer Science Department, Mustanisiryah University, College of Education
[3]Imam Al-kadhum College (IKC), Computer Technique Engineering Department
haidernajem@alkadhum-col.edu.iq

Article History: Received: 10 January 2021; Revised: 12 February 2021; Accepted: 27 March 2021; Published online: 28 April 2021

Abstract: The development of the Web as a highly integrated user interface to real link things has brought many challenges and problems to study, which has led to a rapidly increasing area of research called the Web of things (WoT). Present Web of things (WoT) research is a trigger for the Internet of Things (IoT) growth, opening up opportunities to create ambient areas where people and things interact seamlessly via the Web. This proliferation created concerns among users about the increased usage of Web of Things (WoT) without ensuring that the data generated by its devices are maintained. Many ways to maintain authentication by using lightweight speed algorithms to encrypt and validate the relevant parameters. In the authentication of information, many algorithms have been developed to ensure that the data generated from a physical sensor to the user environment is authentic. These include SHA-1, SHA-2, SHA-3, etc. In Web of Things (WoT), it must be essential to ensure data validity and the continuation of data monitoring to ensure encryption and authentication speed. Therefore, the provision of fast algorithms is an essential requirement for the Web of Things (WoT). This paper includes a modification of the Secure Hash Algorithm 3 (SHA-3) with another high-speed algorithm (Salsa20), which creates a high-speed and secure algorithm in the sensor data validation process. The expanded logistic method would also produce the Secure Hash Algorithm 3 (SHA-3) algorithm's initial values unknown and not identifiable by the intruder. Correspondingly, the National Institute of Standards and Technology (NIST) fifteen statistical tests successfully surpassed the randomness of a proposed method.

Keywords: IoT, WoT authentication, Information system, Security, Salsa20 algorithm, SHA-3

1. Introduction

The next logical move is to use the World Wide Web, and its associating technology to function as an intelligent network as more and more devices are linked to the Internet (i.e., sensor and actuator networks, embedded devices, electronic appliances and digitally enhanced everyday objects) [1]. The Web of Things concept incorporates smart things and their services ultimately into the Web by reusing and adapting technologies and patterns typically used for conventional web content [2]."The Web of Things" can be accomplished by implementing web architecture concepts to integrate real-world objects and built-in devices into the Web seamlessly rather than use the Internet as an infrastructure for transport – as it does for web services. The goal is to make devices part of the network and its infrastructure and resources using HTTP as a protocol on the application layer [3].In the Web of things method, the essential contribution is to provide the basis for the next step beyond fundamental network connectivity. We hope that the Web of Things would do what the Web did for informational resources for real-world resources: simple connectivity was essential but not a necessary condition to make the Internet develop as spectacularly as it still grows today; it was the Web's infrastructure that allowed data and service sharing in ways that had not been seen before [4]. There are two options for connecting intelligent devices to the Web: indirect connectivity and direct connectivity, as seen in fig.1. Home devices can be seen as direct integration with an RFID reader and an embedded server as indirect connectivity. In general, the framework can not only hybridize all integration methods connectivity [5].

Fig.1: WoT overview [5].

As WoT is continuously used, users are worried about the future concerning information security and how data protection and integrity can be protected when moving to the source's final interface. Many people have also avoided using the Internet of Things. There are also various strategies and recommendations for data integrity maintenance [6].

The following paper is organized: section 2 includes a brief overview of applications the Web of Things (WoT). The Challenges in the Web of Things (WoT) security will be presented in section 3. The related work information and cryptographic hash functions (CHF) are presented in section 4 and section 5. The secure hash algorithm-3 (SHA-3) and Salsa20 algorithm are presented in section 6 and section 7. Sections 8 presents the proposed approach system. Section 9 describes the experimental results and discussion. Section 10 ultimately offers a conclusion.

2. Applications of The Web of Things (WoT)

In many real-life applications, WoT is currently used. Experts continuously use these technologies to cater to the community's needs. For this real-world, several WoT applications are creating wonders. Some researchers have used WoT and IoT to improve users' daily lives in specific areas and application areas [7-11]. After analyzing WoT applications in-depth, the authors described in Table 1 the advantages and WoT applications problems.

Mixing IoT and WoT may give many advantages in the Information Technology field. In [12], authors used the mixing of IoT and WoT in robot technology to analyze WoT's actions by using the bibliographic analysis to find many practical examples like sports, fitness, entertaining, society, army, help and surveillance, etc. The general observation is that the researchers have not yet touched on various big WoT/IoT fields in robotics.

The authors in [13] suggested a WoT case study on agriculture. Separate animal monitoring methods and various fixed environmental sensors developed ontology which enabled land performance improvement in architecture. The authors proposed a WoT architecture for vehicle connection via an Interoperable ITS-G5 to share data between vehicles, processing units and storage units [14].

WoT encouraged using similar sensors to evaluate data and light to boost the advantages of reducing costs and human resources through increased flexibility and efficiency in infrastructure operations [15-16]. The management of waste before WoT was a significant problem. To enhance the waste management and collection path, waste containers can then be monitored [17].

The Medical Services Internet has shown epic health benefits, such as disease handling, remote patient control, treatment technologies, and costs and errors. The lives of patients and health staff have been severely affected [18]. Unique intelligent devices and sensors will track the patient's temperature, pulse and other health conditions continuously.

Table1: Real life examples of WoT [19-27].

Application Field	Characteristics	Advantages	Difficulties
Smart infrastructure	Using traffic information. Maintains a clean and healthy resident. Used for emissions prevention.	Improves versatility and confidence. Reduction of workforce. Improvement of the atmosphere by reduction Pollution and noise.	Risk decision-makers are fast and ultra-efficient. Many appropriate handling instruments Data produced. Complexity of applications.
Smart education	Link students from across the globe. Ensure safe and safe	Improving energy efficiency. Pre-efficient. Economic.	Security and confidentiality. Better leadership. Cost reduction.

	Student learning environment. Turn students into designers.	Improves administration and management.	
Agriculture	Soil quality and humidity level control. Watering plans streamlined. Land that fertilizes.	Using less energy to optimize performance. Pre-efficient. Economic. Produces high quality crops. Better climate control for improved crop production.	Ongoing internet connectivity. It is more difficult for the city to embrace the use of technology. Many farmers lack technological expertise.
Healthcare	Usage of intelligent biomedical equipment. Continuous patient surveillance. Medicines customized.	Prevention of diseases and risk surveillance. The saving of expenses. Technological change all over. Early diagnosis and disease prevention.	Protection of personal data sensitive. Daily upgrade difficulties. Management of enormous data. Scalability. Interpretation of results.
Connected cars	Car internet access. Navigation framework Google to Earth.	Ensure security. Regulation of traffic. Sustainability.	It's costly. Security. Infrastructure lack.
Environment monitoring	Air pollution control. Reset of ozone. Moisture, temperate and small dust particles monitoring.	Tsunami and earthquakes are detected early. Air and water monitoring conditions.	Quality of energy
Industries	Maintenance predictive. Scheduling of repairs. Energy skills.	Cost-effective. Manufacturing load optimization.	Tolerance to disappointment. Calculation transmitted.

3. Challenges in The Web of Things (WoT) Security

In terms of privacy, confidence and protection, sharing and transparency are still contradictory. These are the critical challenges in WoT that require exceptional care and more study. Take an example of something available on the Web to understand them. The issue of privacy here only applies to authorized persons accessing this item [28]. The security problem concerns which can and can access the object. The last issue of trust handling was the communication of the various WoT companies on the Web. Many users share and access so that WoT can be successful and famous for the researcher. Unfortunately, researchers have so far mainly addressed IoT problems [29-30]. As a result, challenges to WoT safety have yet to be addressed. In this section, the most demanding safety threats have been identified and summarized in Table 2.

Table 2: Threats of summary related to IoT and WoT [31-37].

Threats	Violated Property	IoT	WoT
Unauthorized access	Authorization	Unauthorized access poses a security vulnerability to the application layer. Suppose multiple users and diverse access rights may exist in an application. The application layer also needs sufficient authentication and access control mechanisms.	Unauthorized WoT users can access a service, software, server and website via someone else's accounts and transfer it to the expected devices, switching the level of various authentication and authority methods. For example, in clever homes, unblocking certain rooms could jeopardize the user's privacy.
Eavesdropping	Authentication	Eavesdropping is a safety hazard to the network layer. When someone attempts to steal information transmitted through the Internet by smartphones, tablets, computers or other devices, each kind of attack begins Sniffing information with some methods, such as packets sniffer.	This is an assault by an individual in the medium, as the traffic flows between the various objects privacy could compromise. By controlling the username, password and number of the credit card.
DoS-attack	Availability	A service that typically fails due to an infrastructure that cannot deal with traffic congestion is often a safety hazard to the network layer. This attack is riskier than the others, as it is effectively enforced on intelligent vehicles, leading to life loss. Mitigation and DoS detection solutions appear to be inadequate and require attention.	DoS is now the most common and most straightforward system attack. They can be configured to minimize the system's speed and capacity to achieve anticipated results in many ways. For example, flood, collision and jamming signals are included in the wireless system. Yahoo, Amazon, CNN and other pages experienced a similar attack in February 2000.
Temp-attacks	Integrity	The assault on tempering is not included in the framework of the Internet of Things	Modify records, memory, or hardware without the customer's knowledge
Impersonating-things	Confidentiality	Refers to it doesn't have anything to do with the IoT.	An intruder behaves like someone else. Impersonate happens. The details can be divulged, or malicious code can be sent.

4. Related Work

Some papers on the WoT implementation of SHA-3 have recently been released. In [38], the proposed implementation of SHA-3 using the LUT high-speed Xilinx Look Up-Table, which would make the Bump support scheme acceptable for WoT applications in the Wire (BITW) security. In [39], proposed using FPGA, appropriate

for sensors WBSN, to implement SHA-3 compact hardware and Carry out a sensor area efficiency assessment and compare our findings with other newly suggested SHA-3 hardware implementation. In [40], SHA was used to overcome the problems of validating information by using two SHA functions and a joint key to generate a hash message authentication code (HMAC). In [41], submitted a report on threats facing IoT users and offered ways to protect data protection and integrity. Usage of hashtags or signatures between parties to protect data integrity prevents its contents from being distorted or changed. In [42], modified SHA-3 used to maintain data integrity in the proposed IoT system. This is done through the use of chaos keys and uses within SHA-3. This is done through two modifications in the algorithm. The first is to rotate the data block to the left by 3 and make XOR with a selected chaos key. Second, it uses the SHA-256 algorithm with a select key switch to produce the KecHash used in the Keccak function. In [43], presents a new secure hash algorithm named (2AMD-160) using the MIT computer science laboratory and RSA data security, Inc.'s famous hash algorithm structure. Compare the procedure with two methods: MD5 and SHA-1 hash feature algorithms to show the efficacy of 2AMD-l60 in security and execution time.

This paper suggested a fast authentication method that would ensure that the used WoT system's information would not be forged and that its content was not changed. Using the modified SHA-3 algorithm with the Salsa20 algorithm, this is done to provide a robust SHA-3 algorithm than the original algorithm to provide quick validation of the system's information.

5. Cryptographic Hash Functions (CHF)

A hash function (HF) is a cryptographic operation that transforms an input message for the variable length to a fixed message called a hash message or message digest. In authentication schemes or random generators, mechanisms for data integrity and digital signatures, hash functionality can be used [44-49].

One of the essential cryptographic hash functions (CHF) is a tool to achieve many safety objectives such as authenticity, numerical signatures, generation of pseudo numbers, digital steganography, and digital time stamping, etc. [50-52].

Many hashing algorithms have emerged that have helped maintain data integrity and authentication; these algorithms were then used to determine the data's validity from the source to the end interface. Therefore, agree on a form of encrypted data with one of the stable hash algorithms between the two parties to ensure the data's confidentiality that can be sent and received between the various systems [53-54].

6. Secure Hash Algorithm-3 (SHA-3)

SHA-3 is selected from the different hashing algorithms to be the default algorithm, based on the Keccak. Keccak has developed the new Secure Hash Algorithm-3. It includes different SHA-3 models such as SHA-3(224-bit, 512-bit, 384-bit and 256-bit) [55]. It includes different rounds and includes logical operations per round. The sponge function is produced to allow input first to be absorbed and then squeezed to produce the desired output [56].

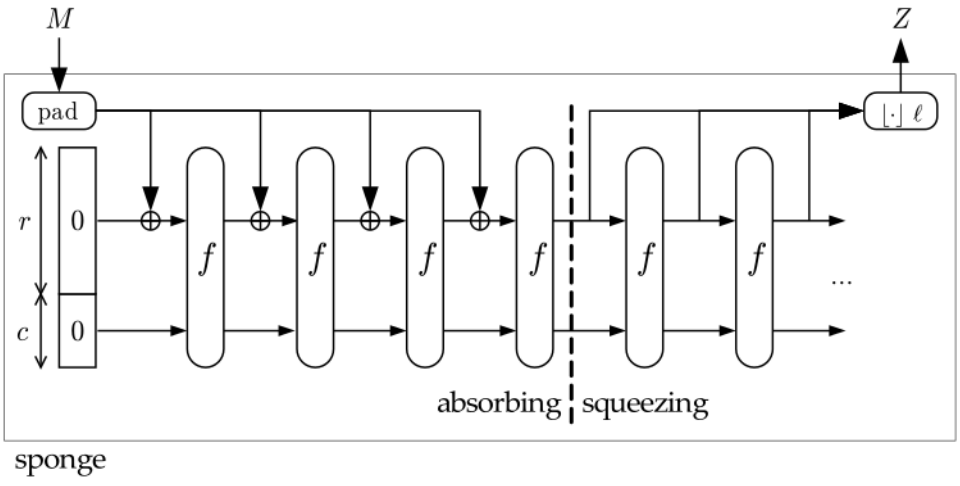

Fig.2: Structure of sponge function [56].

Sponge function consists of three processes: initializing, absorbing and squeezing the input message 'M' as shown in fig.2 and the input hash value as 'Z' on the output side of the padding module [57];

- The zero input step initializes the input matrix, and the input padding is carried out to generate 1600 bits arbitrary input block.
- The input matrix is then XORed in the absorbing process, and all 24 rounds are performed.
- The required output length is achieved by truncating the input matrix during the squeeze process.

7. Salsa20 Algorithm

The Salsa20 algorithm stream cipher was created by Daniel J. Bernstein and submitted to the eStream (ECRYPT Stream Cipher Project) in 2005. It is an asynchronous stream cipher that generates its output as 64-block, blocks extracted from key, nonce, and block number; so, the Salsa20 algorithm enables a separate output block from any original input block everywhere [58]. Salsa20 algorithm pre-processes no S-box and no input, and therefore it calculates an output block that applies the algorithm directly to key input values and nonce input values. It has a 32 byte (256 bits) and 20 rounds built. However, with its 8 and 12 round variants and a minor key, it can be used. The Salsa20 algorithm author does not recommend a little more minor key (16-byte key) as the algorithm doubles it to a 32-byte key [59-60].

The algorithm itself is based on the add-on, XOR and constant distance rotation, all over a 32-bit word array of components (key, nonce, block counter and constant words). The output of this matrix is a 16-word stream used for encryption (XOR) of the plaintext after n-rounds are used [61-62].

8. The Proposed Approach System

The main idea of the proposed approach system is to use the Salsa20 algorithm and SHA-3 as a combination by combing the benefit of each other. In the proposed, all initial values and parameters between the two sides will be defined (sender-side and server-side). The sender-side data sensing enters Salsa20 algorithm for the encryption process. Encrypted data and sensing date move into SHA-3 to produce a hash value more complicated. Then, cipher data and hash value are sent to the server-side, as shown in fig.3. The chips have a 2^8-bit (k0 to k7) key, a 2^6-bit (v0 to v1) nonce and a 2^6-bit counter (n0 to n1). Salsa20's 128-bit key variant doubles the 128-bit key. The other four terms are used for fixing known constants, referred to as c0, c1 and c2 and c3.

Salsa20 algorithm in the proposed approach system has four functions: quarter round, row round, column round, and double round. The central role is quarter-round; double round covers all of the functions and is performed ten times. By XORing a plaintext with the first b bytes of the stream, it encrypts a plaintext of B-byte and rejects the rest of the stream. It decrypts b-byte ciphertext by XORing the ciphertext with the first b bytes on the stream. No input on the stream is available in the plaintext or ciphertext. Salsa20 algorithm generates the stream in 64 bytes (512-bit) blocks. The main, the nonce, and 64-bit block numbers are provided separately for each block. There's no chain between blocks. Thus the random access to the Salsa20 algorithm output stream will parallel any number of blocks.

The received packet will be verified on the server-side by computing the hash using the proposed authentication algorithms on each received packet payload data and comparing the results with the final hash stored in the received packet. Depending on the comparative data, the packet was accepted or rejected. The server side applies the same to all sender-side operations for the final hash-generation, but the final hash-out produced will be compared to the final hash-out received (from sender packet) for each packet received by the server.

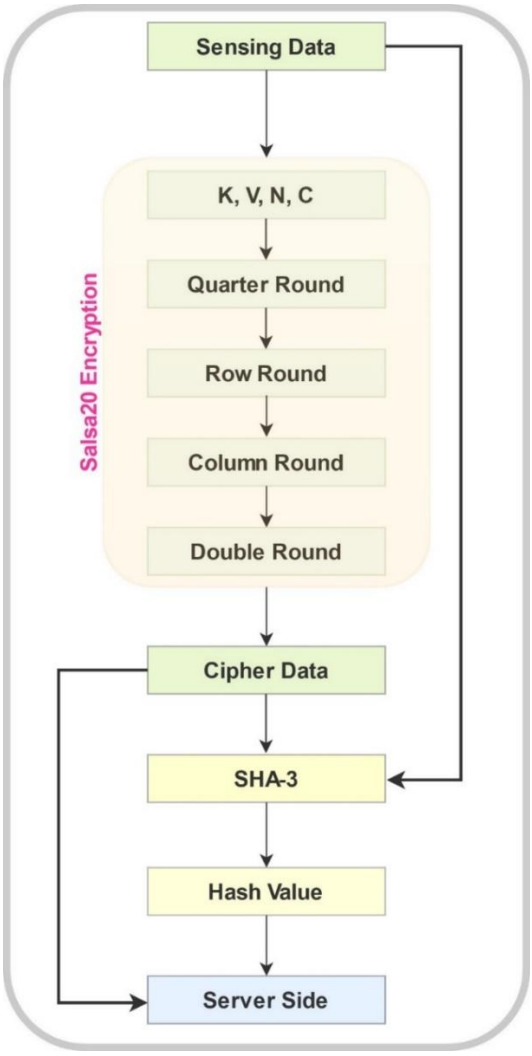

Fig.3: Block diagram of the proposed approach system.

9. Experimental Results and Discussion

This work ensures the encryption and decryption scheme in both software and hardware is straightforward and easy to implement for resource-controlled devices like smart devices and wireless nodes. Text data obtained by sensor devices are the form of data suggested for this proposed approach system. This work examines the possibility of mixing a hash function (SHA-3) and a stream cipher (Salsa20 algorithm) to produce a strong cipher and substantial hash value. The encryption process requires input plaintext data through 20 iterations (rounds), while the decryption process involves reverse of the encryption process. The power consumption is lower, and the encryption speed in the system is more rapid. Our proposal is designed to be used in high security required low-cost devices as it is resistant to most of the cryptanalytic attacks common to stream ciphers. Fifteen tests were developed to assess and quantify the random of binary sequences generated by either software or hardware-based random or pseudo-random number generators for cryptographic applications by the National Institute of Standards and Technology (NIST). There are two approaches to the NIST: to examine the proportion to which a statistical test is performed and to verify uniformity with the distribution of P-values. The NIST tests show that the sequence ratio is random or is not dependent on the value (α) with a default value (0.01). Hence, the sequence is called random if the P-value is as high as 0.01, but the P-value less than 0.01 means the sequence is not random. Table 3 shows all Fifteen NIST experiments in both the proposed approach system and the Salsa20 algorithm and the results for each test. The NIST test results indicate that the proposed approach system's P-values are greater than the P-values of the Salsa20 algorithm for most experiments. As such, there is considerable randomness in the sequence created by the proposed approach scheme.

Table 3: NIST Tests.

Test	P – Value of Salsa20	P – Value of The Proposed
Monobit	0.400110	0.510210
Frequency	0.312074	0.498178
Runs	0.487880	0.587880
Longest Run	0.504688	0.614798
Binary Matrix Rank	0.477522	0.487623
DFT	0.451004	0.651005
Non-Overlapping-Template-Matching	0.679095	0.779095
Overlapping-Template-Matching	0.980653	0.990754
Maurer's Universal	0.323035	0.411046
Linear Complexity	0.514672	0.600674
Serial	0.495961	0.595460
Approximate Entropy	0.520745	0.720565
Cumulative Sums	0.422182	0.518182
Random Excursion	0.399397	0.649397
Random Excursion Variant	0.441425	0.801425

According to time-consuming and complexity applies to the proposed approach in milliseconds, in table 4, the proposed approach indicates, in particular in terms of complexity, that is larger than the standard Salsa20 algorithm approach; however, the approach proposed does require more time to indicate a good indicator of 60 milliseconds, if applicable to real-time apps or to all applications.

Table 4: Comparison between Salsa20 algorithm and the proposed timing and complexity.

Metrics	Standard Salsa20 algorithm	Proposed Approach System
Time	29	34
Complexity	Medium	High

Table 5 presented many factors compared between the original algorithm SHA-3 and the proposed approach system that indicates the proposed have rounds less than the original SHA-3 and high complexity.

Table 5: Comparison between SHA-3 and the proposed approach system.

Factors	SHA-3	Proposed Approach System
Rounds	24 rounds	20 rounds
Complexity	Medium	High
Operation	And, XOR, Rotation, Not	Addition, XOR, Rotation
Memory capacity	More	Less
Block size	256	512

Table 6 shows that when applying any text to the proposed approach with different data changes between numbers and letters, the modified algorithm added more confusion to the clear text with the possibility of verifying the data after applied to it, and this indicates that the proposed approach contains more security features and provides better data distribution density when compared to the original algorithm. Also, if one character is replaced by plain text, this will affect all proposed outputs.

Table 6: The Hashing Time comparison.

Tests	SHA-3	Proposed Approach System
128 Byte	0.001	0.003
256 Byte	0.005	0.008
512 Byte	0.009	0.014
1 Kilo Byte	0.016	0.023
10 Kilo Byte	0.15	0.281

10. Conclusion

IoT is a viable technology to make our lives more comfortable by better responding to emergencies, significantly changing our everyday lives. While it has many benefits, its feature makes it very vulnerable to attacks and seriously affects the application and user. Encryption is rudimentary security that can avoid such attacks, but

it is impossible to use conventional security algorithms because of the limited resources in IoT devices. The proposed approach system for WoT was proposed in this work. The model relies on the Salsa20 algorithm cipher, which was assessed by the global community, which concluded that it was powerful to be part of the eSTREAM and modified SHA-3. The proposed uses the Salsa20 algorithm as a central aspect, but with less input variables,, a lower memory space, and less processing time. Because the information sent by WoT devices is typically contextual data in the form of simple packets, the proposed approach system cipher has a limited output size; thus, no unneeded CPU and memory usage is executed. The performance of the cipher was tested for randomness using the NIST fifteen statistical tests, which were created to evaluate pseudo-random numbers for cryptographic applications and successfully exceeded the randomness of the proposed approach system.

Referenses

1. A. Braden, “Introduction to the Web Of Science,” vol. 2020, 2013.
2. B. Practices, Architecting the Internet of Things, no. May. 2011.
3. M. H. Miraz, M. Ali, P. S. Excell, and R. Picking, “A review on Internet of Things (IoT), Internet of Everything (IoE) and Internet of Nano Things (IoNT),” 2015 Internet Technol. Appl. ITA 2015 - Proc. 6th Int. Conf., no. September, pp. 219–224, 2015, doi: 10.1109/ITechA.2015.7317398.
4. L. Shu, H. H. Chen, T. Hara, D. J. Deng, and L. Wang, “Special issue on recent advance on wireless networks,” J. Commun., vol. 6, no. 6, pp. 421–423, 2011, doi: 10.4304/jcm.6.6.421-423.
5. R. Sardar and T. Anees, “Web of Things: Security Challenges and Mechanisms,” IEEE Access, vol. XX, pp. 1–1, 2021, doi: 10.1109/access.2021.3057655.
6. H. Najm, H. K. Hoomod, and R. Hassan, “A New WoT Cryptography Algorithm Based on GOST and Novel 5d Chaotic System,” Int. J. Interact. Mob. Technol., vol. 15, no. 02, p. 184, 2021, doi: 10.3991/ijim.v15i02.19961.
7. M. R. Faheem, T. Anees, and M. Hussain, “The web of things: Findability taxonomy and challenges,” IEEE Access, vol. 7, pp. 185028–185041, 2019, doi: 10.1109/ACCESS.2019.2960446.
8. M. Kumar, “A Review on Applications of Internet of Things,” Int. J. Adv. Res. Comput. Sci. Softw. Eng., vol. 7, no. 6, pp. 756–760, 2017, doi: 10.23956/ijarcsse/v7i5/0171.
9. P. Goyal, A. K. Sahoo, T. K. Sharma, and P. K. Singh, “Internet of Things: Applications, security and privacy: A survey,” Mater. Today Proc., no. xxxx, 2020, doi: 10.1016/j.matpr.2020.04.737.
10. H. Najm, H. K. Hoomod, and R. Hassan, “Intelligent Internet of Everything (IOE) Data Collection for Health Care Monitor System,” vol. 29, no. 4, pp. 2341–2350, 2020.
11. M. S. Mahdi, “Proposed Secure Internet of Everything (IoE) in Health Care,” 2018.
12. A. Kamilaris and N. Botteghi, “The penetration of Internet of Things in robotics: Towards a web of robotic things,” J. Ambient Intell. Smart Environ., vol. 12, no. 6, pp. 491–512, 2020, doi: 10.3233/AIS-200582.
13. L. S. Zaremba and W. H. Smoleński, “Optimal portfolio choice under a liability constraint,” Ann. Oper. Res., vol. 97, no. 1–4, pp. 131–141, 2000, doi: 10.1023/A.
14. K. Taylor et al., “Farming the web of things,” IEEE Intell. Syst., vol. 28, no. 6, pp. 12–19, 2013, doi: 10.1109/MIS.2013.102.
15. A. Heil, M. Knoll, and T. Weis, “The Internet of things - Context-based device federations,” Proc. Annu. Hawaii Int. Conf. Syst. Sci., no. January, 2007, doi: 10.1109/HICSS.2007.547.
16. S. Fang et al., “An integrated information system for snowmelt flood early-warning based on internet of things,” Inf. Syst. Front., vol. 17, no. 2, pp. 321–335, 2015, doi: 10.1007/s10796-013-9466-1.
17. J. Poncela et al., “Smart cities via data aggregation,” Wirel. Pers. Commun., vol. 76, no. 2, pp. 149–168, 2014, doi: 10.1007/s11277-014-1683-5.
18. F. Alsubaei, A. Abuhussein, V. Shandilya, and S. Shiva, “IoMT-SAF: Internet of Medical Things Security Assessment Framework,” Internet of Things, vol. 8, no. October, p. 100123, 2019, doi: 10.1016/j.iot.2019.100123.
19. J. Mack, Y.-H. (Frank) Hu, and M. A. Hoppa, “A Study of Existing Cross-Site Scripting Detection and Prevention Techniques Using XAMPP and VirtualBox,” Va. J. Sci., vol. 70, no. 3, p. 1, 2019, doi: 10.25778/bx6k-2285.
20. A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, “Internet of things for smart cities,” IEEE Internet Things J., vol. 1, no. 1, pp. 22–32, 2014, doi: 10.1109/JIOT.2014.2306328.
21. M. Abdel-Basset, G. Manogaran, M. Mohamed, and E. Rushdy, “Internet of things in smart education environment: Supportive framework in the decision-making process,” Concurr. Comput. , vol. 31, no. 10, 2019, doi: 10.1002/cpe.4515.
22. B. B. Lin, “Resilience in agriculture through crop diversification: Adaptive management for environmental change,” Bioscience, vol. 61, no. 3, pp. 183–193, 2011, doi: 10.1525/bio.2011.61.3.4.
23. A. A. R. Madushanki, M. N. Halgamuge, W. A. H. S. Wirasagoda, and A. Syed, “Adoption of the Internet of Things (IoT) in agriculture and smart farming towards urban greening: A review,” Int. J. Adv. Comput.

Sci. Appl., vol. 10, no. 4, pp. 11–28, 2019, doi: 10.14569/ijacsa.2019.0100402.
24. S. Tian, W. Yang, J. M. Le Grange, P. Wang, W. Huang, and Z. Ye, "Smart healthcare: making medical care more intelligent," J. Glob. Health, vol. 3, no. 3, pp. 62–65, 2019, doi: 10.1016/j.glohj.2019.07.001.
25. M. Khalaf, H. Najm, A. A. Daleh, A. H. Munef, and G. Mojib, "Schema Matching Using Word-level Clustering for Integrating Universities' Courses," in 2020 2nd Al-Noor International Conference for Science and Technology (NICST), 2020, pp. 1–6.
26. M. S. Mahdi, "Computer Aided Diagnosis System for Breast Cancer using ID3 and SVM Based on Slantlet Transform," Qalaai Zanist J., vol. 2, pp. 142–148, 2017.
27. M. S. Mahdi and N. F. Hassan, "A Proposed Lossy Image Compression based on Multiplication Table," Kurdistan J. Appl. Res., vol. 2, no. 3, pp. 98–102, 2017.
28. H. HaddadPajouh, A. Dehghantanha, R. Khayami, and K. K. R. Choo, "A deep Recurrent Neural Network based approach for Internet of Things malware threat hunting," Futur. Gener. Comput. Syst., vol. 85, no. November, pp. 88–96, 2018, doi: 10.1016/j.future.2018.03.007.
29. H. Najm, H. Ansaf, and O. A. Hassen, "AN EFFECTIVE IMPLEMENTATION OF FACE RECOGNITION USING DEEP CONVOLUTIONAL NETWORK," J. SOUTHWEST JIAOTONG Univ., vol. 54, no. 5, 2019.
30. H. K. Tayyeh, M. S. Mahdi, and A. S. A. AL-Jumaili, "Novel steganography scheme using Arabic text features in Holy Quran," Int. J. Electr. Comput. Eng., vol. 9, no. 3, p. 1910, 2019.
31. I. Ali, S. Sabir, and Z. Ullah, "Internet of things security, device authentication and access control: A review," arXiv, no. August, 2019.
32. Y. Ashibani and Q. H. Mahmoud, "A User Authentication Model for IoT Networks Based on App Traffic Patterns," 2018 IEEE 9th Annu. Inf. Technol. Electron. Mob. Commun. Conf. IEMCON 2018, pp. 632–638, 2019, doi: 10.1109/IEMCON.2018.8614892.
33. A. K. Farhan and M. S. Mahdi, "Proposal Dynamic Keys Generator for DES algorithms," Islam. Coll. Univ. J., vol. 29, pp. 25–48, 2014.
34. A. Kadhim and M. Salih, "Proposal of New Keys Generator for DES Algorithms Depending on Multi Techniques," Eng. Technol. J., vol. 32, no. 1 Part (B) Scientific, pp. 94–106, 2014.
35. A. F. Majeed and I. A. Murdas, "Novel design for li-fi healthcare monitoring system," Int. J. Intell. Eng. Syst., vol. 12, no. 6, pp. 59–70, 2019, doi: 10.22266/ijies2019.1231.06.
36. A. N. Jabbar and I. A. Murdas, "Developing a graphical package for sensor arrays design, optimization and maintenance," Int. J. Eng. Technol., vol. 7, no. 3, pp. 1388–1399, 2018, doi: 10.14419/ijet.v7i3.14113.
37. G. H. Eskandar, A. N. Jabbar, and I. A. Murdas, "Building a Noisy Multipath Channel Emulator for Single or Multicarrier Communication Systems," in 2019 International Conference on Advanced Science and Engineering (ICOASE), 2019, pp. 147–152.
38. M. Rao, T. Newe, I. Grout, and A. Mathur, "High speed implementation of a SHA-3 core on virtex-5 and virtex-6 fpgas," J. Circuits, Syst. Comput., vol. 25, no. 07, p. 1650069, 2016.
39. Y. Yang, D. He, N. Kumar, and S. Zeadally, "Compact Hardware Implementation of a SHA-3 Core for Wireless Body Sensor Networks," IEEE Access, vol. 6, no. 1, pp. 40128–40136, 2018, doi: 10.1109/ACCESS.2018.2855408.
40. M. Z. A. Bhuiyan, T. Wang, T. Hayajneh, and G. M. Weiss, "Maintaining the balance between privacy and data integrity in internet of things," in Proceedings of the 2017 international conference on management engineering, software engineering and service sciences, 2017, pp. 177–182.
41. B. Baldwin et al., "A hardware wrapper for the SHA-3 hash algorithms," 2010.
42. G. H. A.-M. and A. K. F. Jolan Rokan Naif Al-Khazraji, "Design And Implementation Of Secure IoT for Emergency Response System Using Wireless Sensor Network and Chaotic," 2019.
43. H. M. Al-Mashhadi, H. B. Abdul-Wahab, and R. F. Hassan, "Secure and time efficient hash-based message authentication algorithm for wireless sensor networks," GSCIT 2014 - Glob. Summit Comput. Inf. Technol., 2014, doi: 10.1109/GSCIT.2014.6970116.
44. R. Chaves et al., "Secure hashing: SHA-1, SHA-2, and SHA-3," Circuits Syst. Secur. Priv., pp. 81–107, 2017, doi: 10.1201/b19499.
45. I. Jabbar and S. N. Alsaad, "Design and implementation of secure remote e-voting system using homomorphic encryption," Int. J. Netw. Secur., vol. 19, no. 5, pp. 694–703, 2017, doi: 10.6633/IJNS.201709.19(5).06.
46. I. Jabbar, "Using Fully Homomorphic Encryption to Secure Cloud Computing," Internet Things Cloud Comput., vol. 4, no. 2, p. 13, 2016, doi: 10.11648/j.iotcc.20160402.12.
47. A. A. K. Jaber, S. J. Mohammed, and I. A. Murdas, "Simulation of a SAC-OCDMA 15 User System Using Hadamard Code," NICST 2019 - 1st Al-Noor Int. Conf. Sci. Technol., no. Nicst, pp. 38–41, 2019, doi: 10.1109/NICST49484.2019.9043796.
48. I. A. Murdas, "Quadrature amplitude modulation all optical orthogonal frequency division multiplexing-

dense wavelength division multiplexing-optical wireless communication system under different weather conditions," Int. J. Eng. Trans. A Basics, vol. 30, no. 7, pp. 988–994, 2017, doi: 10.5829/ije.2017.30.07a.08.

49. S. M. Hamzah and I. A. Murdas, "Enhancement of the performance of DWDM free space optics (FSO) communications systems under different weather conditions," Int. J. Intell. Eng. Syst., vol. 13, no. 4, pp. 446–456, 2020, doi: 10.22266/IJIES2020.0831.39.
50. R. Sobti and G. Geetha, "Cryptographic Hash functions - a review," IJCSI Int. J. Comput. Sci. Issues, vol. 9, no. 2, pp. 461–479, 2012.
51. M. S. Mahdi and N. F. Hassan, "Design of keystream Generator utilizing Firefly Algorithm," J. Al-Qadisiyah Comput. Sci. Math., vol. 10, no. 3, pp. 91–99, 2018, doi: 10.29304/jqcm.2018.10.3.441.
52. H. Ansaf, H. Najm, J. M. Atiyah, and O. A. Hassen, "IMPROVED APPROACH FOR IDENTIFICATION OF REAL AND FAKE SMILE USING CHAOS THEORY AND PRINCIPAL COMPONENT ANALYSIS," J. SOUTHWEST JIAOTONG Univ., vol. 54, no. 5, 2019.
53. H. K. Hoomod, J. R. Naif, and I. S. Ahmed, "Modify Speck-SHA3 (SSHA) for Data Integrity in Wot Networking Based on 4-D Chaotic System," Period. Eng. Nat. Sci., vol. 8, no. 4, pp. 2379–2388, 2020, doi: 10.21533/pen.v8i4.1743.
54. A. K. Farhan and M. Ali, "Database protection system depend on modified hash function," in Conference of Cihan University-Erbil on Communication Engineering and Computer Science, 2017, p. 84.
55. L. Henzen, P. Gendotti, P. Guillet, E. Pargaetzi, M. Zoller, and F. K. Gürkaynak, "Developing a hardware evaluation method for SHA-3 candidates," in International Workshop on Cryptographic Hardware and Embedded Systems, 2010, pp. 248–263.
56. J. Sharma and D. Koppad, "Low power and pipelined secure hashing algorithm-3(SHA-3)," 2016 IEEE Annu. India Conf. INDICON 2016, vol. 3, pp. 0–4, 2017, doi: 10.1109/INDICON.2016.7839059.
57. J. James, R. Karthika, and R. Nandakumar, "Design & Characterization of SHA 3- 256 Bit IP Core," Procedia Technol., vol. 24, pp. 918–924, 2016, doi: 10.1016/j.protcy.2016.05.184.
58. P. A. Nikolov, "Analysis and Design of a Stream Cipher," 2019.
59. M. Salih Mahdi and N. Flaih Hassan, "a Suggested Super Salsa Stream Cipher," Iraqi J. Comput. Informatics, vol. 44, no. 2, pp. 1–6, 2018, doi: 10.25195/2017/4422.
60. H. Najm, H. K. Hoomod, and R. Hassan, "A proposed hybrid cryptography algorithm based on GOST and salsa (20)," Period. Eng. Nat. Sci., vol. 8, no. 3, pp. 1829–1835, 2020, [Online]. Available: http://pen.ius.edu.ba/index.php/pen/article/view/1619.
61. M. S. Mahdi, R. A. Azeez, and N. F. Hassan, "A proposed lightweight image encryption using ChaCha with hyperchaotic maps," no. November, 2020.
62. M. S. Mahdi, N. F. Hassan, and G. H. Abdul-Majeed, "An improved chacha algorithm for securing data on IoT devices," SN Appl. Sci., vol. 3, no. 4, pp. 1–9, 2021.