

IoTES (A Machine learning model) Design dependent encryption selection for IoT devices

1st Matasem Saleh

*School of Computer Science and Engineering, SCE,
Taylor's University, Malaysia*
matasemsaleh@gmail.com

3rd Azween Abdullah

*School of Computer Science and Engineering, SCE,
Taylor's University, Malaysia*
azween.abdullah@taylors.edu.my

2nd NZ Jhanjhi

*School of Computer Science and Engineering, SCE,
Taylor's University, Malaysia*
noorzaman.jhanjhi@taylors.edu.my

4th Raazia Saher

*College of Computer Science and Information Technology (CCSIT)
King Faisal University, Saudi Arabia*
raaziasaher@gmail.com

Abstract— The data collected by IoT devices has a tremendous impact on today's world. Not only does it increase one's awareness about his surroundings, but it also enables the businesses to provide a variety of lucrative and personalized services for each end user. The use of IoT technology can combat variety of current as well as future issues which can arise because of urbanization. The expansion of IoT technology requires trust from the end user. However, IoT devices are vulnerable to cyber threats due to storing majority of its collected data in third-party databases. The end users would never like to compromise their confidential data as a result of using this technology. If these devices can ensure user data privacy, then it will increase the end user's confidence and hence lead to a greater adoption of IoT technology. Encryption of IoT data within the device itself builds user trust and eliminates the fear of storing IoT data in a third-party database. The designers of this technology face considerable difficulties in selecting an appropriate encryption technique for the IoT device due to several reasons among which IoT's resource-constrained nature is worth mentioning. This paper presents a model developed by using machine learning to assist IoT device designers in selecting an appropriate encryption for their devices.

Keywords—IoT, System Security, IoT Device Security, Cryptography, Machine Learning, System Design.

I. INTRODUCTION

IoT refers to billions of devices that are connected through internet and able to store and share data worldwide. Due to the cheap cost of computer chip production and the widespread use of communication technologies [1, 2], the number of IoT devices continues to expand worldwide. Statista predicts that, 75,440 billion devices will be operational by 2025 [3].

IoT devices are categorized as heterogeneous devices because they are required to perform optimally in diverse environments for different objectives. By contrast, the context in which IoT devices may be deployed has constraints on their size and resources such as processing power, memory, and energy. Although, IoT devices have variety of components and design architecture, they share key characteristics, such as the ability to gather and communicate data while operating on

minimal resources [4-6]. Sensors are one of the physical components of IoT devices that play a key role in gathering real-world data. The data collected by sensors is sent by the IoT device to distant facilities for analysis. The restricted nature of IoT devices, necessitates the inclusion of a compressed functioning software in such devices. As a result, the operating software in such devices must be lightweight in order to adhere to resource limits.

Because IoT devices are mainly developed to gather and transmit data over the internet, they might be exposed to the traditional cyber threats as any other devices communicating over the internet. However, protecting IoT devices is more challenging due to their limited resources.

The data acquired by IoT devices contains a substantial amount of personal information about the user's device which calls for the need of secrecy from the user's viewpoint. Attackers see the data collected by IoT devices as primary means of gaining access to user information[7]. Indeed, the increased demand for IoT technology has drawn hackers' attention to manipulating and monitoring these devices and the data they collect. To avoid this, IoT devices must be secured against potential threats.[8, 9]. Encrypting the data transmitted by these IoT devices is vital, keeping in mind the worth of data [10]. Therefore, it is a must and strongly recommended practice. It increases users' trust and confidence in IoT technologies by assuring that they do not divulge the acquired data to any unauthorized party. This raise in users' trust will result in an increased social adoption. For a variety of reasons, IoT device designers struggle to find an encryption solution that is suited for their device and performs the desired duty without jeopardizing the device's effectiveness.

To summarize the complexity of this issue, the restrained nature of IoT devices makes the selection of software encryption a challenging task. The designers of IoT devices are accountable for creating a safe and secure product without compromising performance. In this article, we will briefly explain the reasons that contribute to device designers' difficulty in encrypting the messages sent by their devices and present a model to assist them in overcoming these obstacles.

A. Research Contribution

User and environmental data are collected by IoT devices.

Hence, it is critical that IoT data be transferred securely. It boosts customer confidence in IoT applications, which in turn will lead to a greater adoption of this technology in society. IoT device designers have several challenges when it comes to software-level data security in an IoT device, as detailed in article [11] which has already been published. Some kind of support given by various organizations and researchers to aid IoT device designers has been discussed in details and in our previous publication [12]. In addition, assistance must be extended to IoT device designers to help them achieve the maximum security with least effort. Thus we are discussing the results of our previously proposed model [11] in this article.

B. Paper Organization

At the outset of this article, we'll take a short look at the problems designers encounter when selecting an appropriate encryption for their device. Following that, we'll explore the help supplied by various organizations and publications to overcome these difficulties. Later, we'll explore the model we developed and its outcomes.

II. CHALLENGES

In this section, we'll look at the roadblocks that IoT device designers encounter and the reasons why they struggle while trying to decide on which encryption to employ.

The primary objective of IoT devices is to collect real-time data from their surroundings for subsequent analysis. Therefore, IoT devices have been able to be used in a range of environments and domains [13], such as healthcare [14-16], natural disaster [17], retail sector [18], education [19], electricity and transportation management [20-23]. The introduction of IoT across a variety of industries has resulted in a wide range of standards for how IoT devices are used thus creating a challenge for uniformity. The absence of standardization is one of the problems restricting the selection of an encryption approach and precludes labelling the appropriate encryption technology for IoT devices [24-26].

IoT devices are small in size and may be easily integrated into existing systems [27] or as a standalone device such as CCTV cameras. The type of sensor or actuator employed, varies according to the purpose for which the IoT device is being used. It is difficult for IoT devices to work at their full potential if they are integrated into other systems, which restricts their ability to perform as conventional computing devices. A huge range of applications and environments utilize IoT devices. However, the complication is that each IoT device requires a unique encryption technique due to its basic design and components. [28, 29].

Apart from the dramatic increase in the number of IoT devices and their deployment in several industries, which are intended to fulfill certain IoT forecasts, they have had an overwhelmingly favourable influence on the global economy. However, there is a lot of perplexity in the price drop associated with the rising use of IoT. As a result of this price fall, IoT device developers will be less inclined to engage in security research. A crucial factor in the success of IoT device designers is their expertise in embedded systems, which due to the rise of innovative IoT devices has seen rapid development in recent years. Engineers with experience in

embedded systems are often more knowledgeable about optimizing hardware performance and place a lower premium on security [30].

In addition to the difficulties outlined above, the diversity of encryption techniques adds another layer of complexity. Encryption is a method used to protect information from those who are not authorized to view or modify it [31]. It all started with the "Caesar Cipher" and has now refined to more complex algorithms which are secure and unlikely to be breached. IoT devices may be protected using a variety of encryption techniques. Encryption algorithms belong to different core classes i.e. symmetric, asymmetric, lightweight and combined encryption. The symmetric encryption algorithms can be block or stream cipher [32], while asymmetric encryption algorithms can be used either for data encryption or digital signature [33]. Lightweight encryption algorithms are being introduced to be eligible to run on IoT devices by minimizing the encryption algorithm complexity while compromising the level of security [34, 35]. Although, the combined encryption method is more complex but it increases the security level by combining two or more encryption algorithms[12]. The earlier mentioned classes include several underlying encryption algorithms. Each one of them have their own distinct advantages and disadvantages depending on the device's processing capability, memory, and power supply. Selection of an effective encryption methodology from a broad range of available possibilities involves extensive study to predict the encryption technique's impact on IoT device performance. Furthermore, since each encryption strategy provides a different degree of security. Consequently, selecting the encryption technique that places least load on the device may not give the desired level of security. Many of the ongoing researches are providing this kind of analysis and some of them are addressed in the following section.

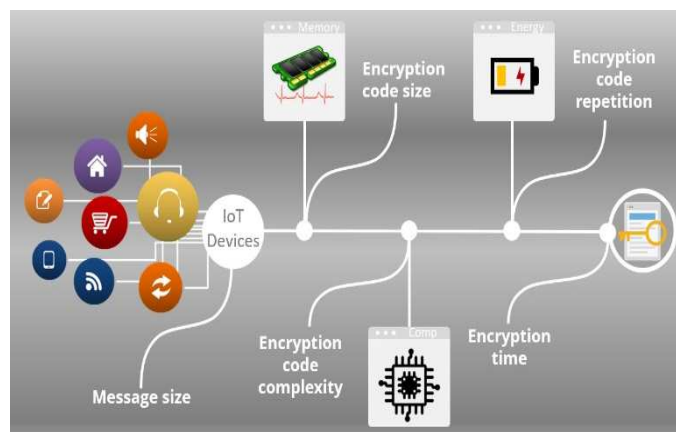


Figure 1. Encryption Code Impact on IoT Device Parameters[11].

III. AVAILABLE SUPPORT

The challenge for IoT device designers is to select an encryption technique that is suitable for their device and does not impose a significant overhead that might have a negative impact on their device. The preceding section examined the reasons behind the task's difficulty. Consequently, several

studies have been published to give assistance by expanding the designer's understanding of encryption techniques and their exploitation of device resources. Published studies conducted numerous tests, analyses, and comparisons of different encryption techniques and presented their findings to increase designers' knowledge and simplify their encryption selection task. In one of the publications, raspberry pi and Arduino were used to evaluate a variety of symmetric encryption algorithms and their impact on the energy consumption of these devices [36]. Another article stated that algorithms such as RSA, 3DES, AES, blowfish, and RC6 are incompatible with IoT devices because of their greater energy consumption than expected [37]. However, given the heterogeneity of the IoT environment, similar encryption approaches might be employed in specific IoT devices with access to the power supply, such as those seen in smart home applications. While another study assessed the efficacy and implementation of four separate symmetric encryption approaches and three distinct asymmetric ones [4]. Another investigation extracted data from the literature on symmetrical and asymmetric encryption and then connected the block and stream to different key sizes [38]. This research discovers that block cipher encryption algorithms are more versatile than stream ciphers. In one publication [39], the Raspberry Pi and Beagle Bone Black were used to compare the execution times of numerous symmetric encryption-decryption approaches and several block and stream ciphers. This study revealed that the Raspberry pi's encryption execution time was quicker. In [40], AES and XTEA were compared. The purpose of this research was to ascertain the feasibility of employing XTEA on embedded low-resource systems.

On the other hand, some organizations have taken the initiative to provide the required knowledge to IoT device designers and underline the critical nature of securing their devices by providing the necessary security without compromising on functionality. The primary objective of these organizations is to encourage and assist IoT device designers to adopt necessary security procedures in order to produce secured IoT devices[41]. Numerous organizations provide guidelines to educate designers and developers on recommended practices for designing safe devices. IoT Security Foundation [42] is a non-profit organization dedicated to the highest standards of IoT protection. There is a lengthy explanation of encryption and the protocols to follow when deploying encryption for an IoT device in their release of Best Practice Recommendations [43].

Infocom Media Development Authority IMDA [44] annually publishes a guideline for designers and developers of IoT devices, as well as for IoT consumers [45]. A discussion on cryptography and how to choose a safe encryption method was included in their January 2019 guideline. Many other organisations have the same objective and perspective as the earlier mentioned examples such as European Union Agency for Cybersecurity (Anisa).

IV. IOTES MODEL IMPLEMENTATION

This section will discuss the implementation of the IOTES

model which we proposed in a previous study. IOTES model was designed as a continuation of the other publication work that was presented in available support topic. The same methods as in the published studies were followed. It includes conducting an experiment to collect the load values for various encryption techniques on many devices' attributes such as memory, CPU and the time was consumed to encrypt a message. The acquired data was compiled into a single dataset for further analysis. The additional step performed in this study is to input this dataset to a machine learning model which will provide dynamic assistance that may recommend an appropriate encryption for each device depending on its specification and the needs of the designer.

A. Experimental Setup.

First step was to select the encryption we wanted to test in our work and include in the dataset. We selected ten different symmetric encryption algorithms, ten different lightweight encryption algorithms and eight asymmetric encryption algorithms. We have 28 total encryption algorithms belonging to three distinct classes. The encryption techniques were written in Python using well-known and reliable libraries. Several key sizes were utilized with each encryption to get a better understanding of the encryption algorithms' behaviour and to determine the effect of a larger key size on device behaviour.

Subsequently, we needed to choose the devices for our experiment. We ultimately chose three devices: a Raspberry Pi 3 model B, a Pocket Beaglebone, and a Raspberry Pi Zero, as shown in the figure 3. All the devices run on Linux, and we chose them in this manner to facilitate data collection, since the goal was to build a single testing framework compatible with all devices. The encrypted messages were of 5 sizes (2, 4, 8, 16 and 32 bytes) and were generated automatically through a function built for that purpose.

	Raspberry_Pi	Pocket	Pi_Zero
Core	4	1	
CPU_Architecture	armv7l		armv6l
CPU_Model	Cortex-A53	Cortex-A8	ARM1176
Frequency	1200	1000	700
RAM_Size	1024	512	
RAM_Type	SDRAM	DDR3	SDRAM

Figure 2. Used devices specifications.

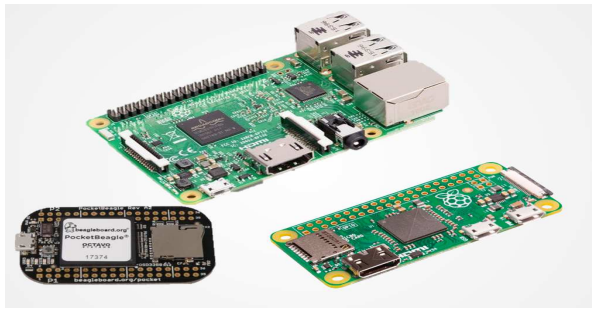


Figure 3. Used devices.

After that, we had to choose a profiler. Many profilers were tested, including `getrusage` python package, which retrieves used data, and the Linux utility command `ps`, which displays information about current processes. Profilers' data are gathered and will be prepared for further studies throughout the analysis phase.

It required more than 60 hours of testing to obtain data from all devices and all algorithms, which included a wide range of key and message sizes.

We built our framework in such a manner that keys are produced dynamically depending on the key size specified in the framework's `config.json` file. The framework captures all data for confirmation purposes, including the input message, key, encrypted message, and decrypted message. In addition, three functions were created: one for key creation, one for message encryption, and last function for message decryption. The time taken and the utilized resources were recorded for each previous function independently.

B. Dataset Description.

The dataset was compiled by conducting multiple tests of encryption algorithms belonging to three distinct encryption types (Symmetric, Asymmetric, and Lightweight) on three distinct devices, each with multiple sets of keys and a range of message sizes. The test was repeated several times. Finally, we collected 37,278 data points, which correspond to three instances (get key, encrypt, and decrypt), as described in the experimental setup. As a result, one-third of our main dataset is considered in this article, which includes the encrypted instance, for a total of 12,426 data points in the dataset that we will analyse. The total number of asymmetric datapoints is 3450, which is nearly a quarter of the total dataset and less than the total number of other datapoint types. In comparison, the other two categories account for about 36% of each. Algorithm column has 28 unique algorithms. while our dataset has a width of 22 columns. Seven columns are related to the encryption algorithms parameters and details which are shown in figure 4. Other seven columns are containing devices specifications data. While five columns related to CPU readings and one column containing the time spent during the encryption and one column has the percentage of the utilized memory. There is one column which contains the CPU temperature measured during the conducted encryption. This temperature was determined from a temperature sensor preinstalled in devices.

Device_Info						
Device_Name	CPU_Architecture	CPU_Model	CPU_Core	CPU_Frequency	RAM_Size	RAM_Type

Encryption					
Type	Algorithm	Functionality	Cipher_Type	Block_Size	Key_Size
Message_Size					

CPU				
CPU%	Performance	CPU_Time	CPU_User_Time	CPU_System_Time

Temperature	Memory Readings	Duration
Temp	MEM_USAGE(%)	Elapsed

Figure 4. Dataset Feature.

C. Data Analysis.

It was necessary for us to analyse the dataset in order to gather important information which we may use to test our model in the following section. We were also able to identify the parameters that our model user would have to choose from based on our dataset analysis. We began by conducting a systematic comparison of the devices' performance for each encryption class to identify the fastest device. To determine which algorithms were most efficient, we looked at the amount of memory, CPU, and execution time they required. Figure 5 demonstrates that `xxtea` has the lowest CPU load in the lightweight class, whereas `clefia` has the most. The analysis has been done inside each encryption algorithm as well by comparing various used keys. As can be seen in the graph, employing the `emo-1` encryption technique with a 64-bit key size results in much quicker processing times, but at the expense of significantly reduced security which is not being considered in our study at this stage. Whereas `RSA` has comparable processing time with 128- and 256-byte keys.

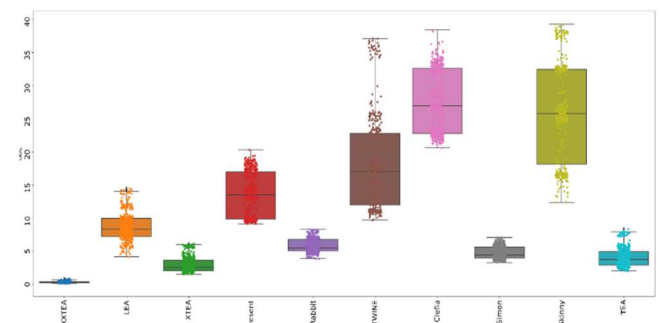


Figure 5. CPU utilization of lightweight encryption algorithms.

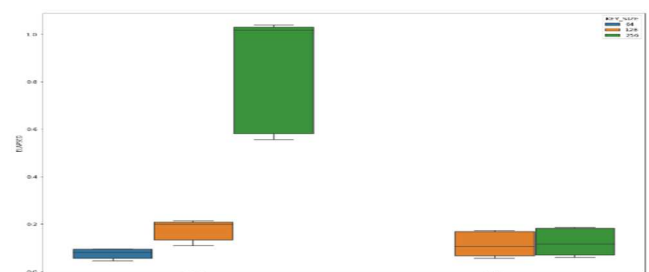


Figure 6. Processing time comparison of RSA and EMO-1 with different key sizes.

Message size influences the algorithm processing time, which is something we've investigated further. To do this, as

seen in the figure 7, we distributed all the data points in our dataset into bands with the equal number of data points and frequencies. As far as we could tell, the algorithms were 18% quicker for messages encrypted with 8 or fewer bytes of data than those with 8 or more bytes of data.

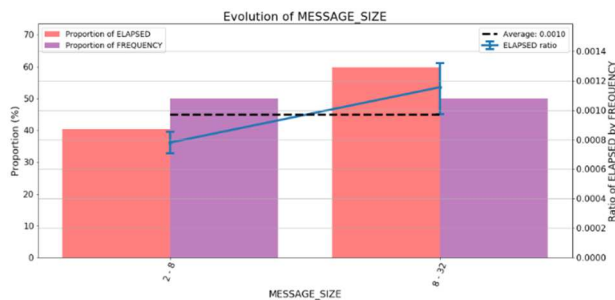


Figure 7. Message size effect on processing time.

D. Data pre-processing and Building Model

Data pre-processing is an essential step in data management before using the data in a machine learning model. This includes cleaning, normalization, and the change of features. The study's categorical features required a great deal of data manipulation. Predictive models will have better performance by converting the categorical data to numeric-categorical values. One hot encoding technique was used to encode the features while label encoding technique was used to transform the target column. Data scaling was necessary for several features as the machine learning algorithms determine the distance between data. The numerical columns were scaled in a way so that the data ranges between [0,1]. Standard scaler is not being used, because the best performing machine learning models are tree-based models and standardization did not have any effect on the results.

For the purposes of testing and training, we divided the data into two sets, comprising of 20% and 80% of the total dataset volume. Data validation was performed on a quarter of the training set to validate the model stability.

We deployed and compared five classification models as shown in table 1. All tree-based models outperform the logistic regression model in terms of accuracy, precision, recall and F1 score, as demonstrated in the following table.

	Type	Logistic Regression	Decision Tree	Random Forest	Gradient Boosting	XGBoost
Accuracy	Train	0.77722	0.99716	0.954649	1	0.983621
	Test	0.762876	0.950107	0.938841	0.964592	0.967275
Precision	Train	0.808221	0.997239	0.961658	1	0.984512
	Test	0.768228	0.997151	0.954912	1	0.983754
Recall	Train	0.745596	0.997177	0.95428	1	0.983741
	Test	0.789412	0.952185	0.950071	0.966572	0.969661
F1 score	Train	0.762165	0.951758	0.941623	0.965388	0.968519
	Test	0.741395	0.951727	0.939078	0.965661	0.96837

Table 1. Models' performance comparison.

While the remaining outcomes are comparable, we plotted

their scores using a boxplot to choose which one to adopt. Although, both gradient boosting and XGBoost may be used because of their excellent performance, we chose the gradient boosting model due to its high precision. Gradient Boosting's confusion matrix indicates that there is a 21% chance that the model will misclassify DES and display it as CAST, even though both are symmetric encryption, and a 12% chance that the model will misclassify XTWINE and display it as PRESENT, with a slightly lower percentage in reverse.

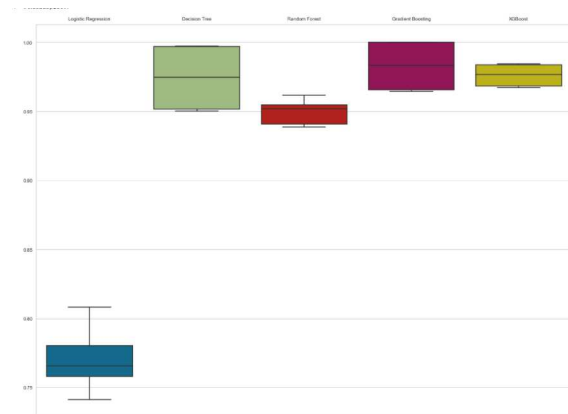


Figure 8. Plotting the results of Machine learning Model.

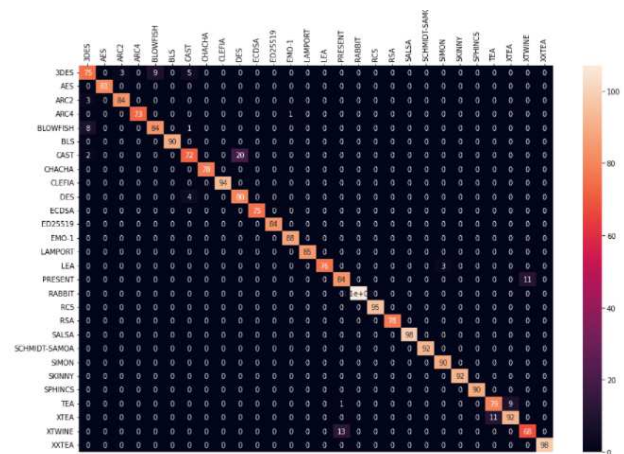


Figure 9. Gradient Boosting confusion matrix.

V. CONCLUSION

Protecting user data is critical for any technology, but especially for those that collect data about people and their surroundings, such as Internet of Things devices. Due to the difficulties experienced by the designers of these devices in terms of balancing security and device performance optimization, the researcher and concerned organizations placed a premium on the security of IoT devices and the data they gather. Facilitating this task with the latest technology might provide extra support and save a lot of time and effort of the designers in picking the most suitable encryption for their device with minimal overhead on device resources. If an appropriate dataset is built, machine learning would be a significant advancement in this area and would simplify the task at hand. In this paper we tested the feasibility of using

machine learning in this field and we discovered that it has an outstanding result especially by using tree-based models.

VI. FUTURE WORK

Extending this work and including additional attributes such as power consumption and temperature will be a significant step toward achieving the goal of providing comprehensive support to designers and developers of IoT devices. Creating a web application that allows designers to interact with the machine learning model and study the suggested algorithms based on their input parameters could be the final step in achieving the goal that we are aiming to implement in the near future.

VII. REFERENCES

- [1] A. Ragab, G. Selim, A. Wahdan, and A. Madani, "Robust Hybrid Lightweight Cryptosystem for Protecting IoT Smart Devices," *Security, Privacy, and Anonymity in Computation, Communication, and Storage*, Lecture Notes in Computer Science, pp. 5-19, 2019.
- [2] S. J. Hussain, M. Irfan, N. Jhanjhi, K. Hussain, and M. Humayun, "Performance Enhancement in Wireless Body Area Networks with Secure Communication," *Wireless Personal Communications*, pp. 1-22, 2020.
- [3] S. H. Mahmud, L. Assan, and R. Islam, "Potentials of internet of things (IoT) in malaysian construction industry," *Annals of Emerging Technologies in Computing (AETiC)*, Print ISSN, pp. 2516-0281, 2018.
- [4] M. N. B. Anwar, M. Hasan, M. M. Hasan, J. Z. Loren, and S. T. Hossain, "Comparative Study of Cryptography Algorithms and Its' Applications," *International Journal of Computer Networks and Communications Security*, vol. 7, no. 5, pp. 96-103, 2019.
- [5] T. Poongodi, R. Krishnamurthi, R. Indrakumari, P. Suresh, and B. Balusamy, "Wearable Devices and IoT," *A Handbook of Internet of Things in Biomedical and Cyber Physical System*, pp. 245-273: Springer, 2020.
- [6] N. Maryanti, R. Rohana, and M. Kristiawan, "The Principal's Strategy In Preparing Students Ready To Face the Industrial Revolution 4.0," *International Journal of Educational Review*, vol. 2, no. 1, pp. 54-69, 2020.
- [7] M. Humayun, N. Jhanjhi, A. Alsayat, and V. Ponnusamy, "Internet of things and ransomware: Evolution, mitigation and prevention," *Egyptian Informatics Journal*, 2020.
- [8] N. Jhanjhi, and A. Alhumam, "Detection and Mitigation of RPL Rank and Version Number Attacks in the Internet of Things: SRPL-RP," *Sensors (Basel, Switzerland)*, vol. 20, no. 21, 2020.
- [9] Z. A. Almusaylim, A. Alhumam, and N. Jhanjhi, "Proposing a Secure RPL based Internet of Things Routing Protocol: A Review," *Ad Hoc Networks*, vol. 101, pp. 102096, 2020.
- [10] M. S. Mahdavejad, M. Rezvan, M. Barekatain, P. Adibi, P. Barnaghi, and A. P. Sheth, "Machine learning for Internet of Things data analysis: A survey," *Digital Communications and Networks*, vol. 4, no. 3, pp. 161-175, 2018.
- [11] M. Saleh, N. Jhanjhi, A. Abdullah, and R. Saher, "Proposing Encryption Selection Model for IoT Devices Based on IoT Device Design." pp. 210-219.
- [12] M. Saleh, N. Jhanjhi, A. Abdullah, and R. Saher, "Design Challenges of Securing IoT Devices: A survey."
- [13] M. Nardelli, S. Nastic, S. Dustdar, M. Villari, and R. Ranjan, "Osmotic flow: Osmotic computing+iot workflow," *IEEE Cloud Computing*, vol. 4, no. 2, pp. 68-75, 2017.
- [14] A. Haleem, M. Javaid, and I. H. Khan, "Internet of things (IoT) applications in orthopaedics," *Journal of Clinical Orthopaedics & Trauma*, vol. 11, pp. S105-S106, 2020.
- [15] S. Jacob, M. Alagirisamy, V. G. Menon, M. Kumar, N. Jhanjhi, V. Ponnusamy, P. Shynu, and V. Balasubramanian, "An Adaptive and Flexible Brain Energized Full Body Exoskeleton with IoT Edge for Assisting the Paralyzed Patients," *IEEE Access*, 2020.
- [16] R. Saher, and M. Anjum, "Role of technology in COVID-19 pandemic," *Researches and Applications of Artificial Intelligence to Mitigate Pandemics*, pp. 109-138: Elsevier, 2021.
- [17] M. Saleh, N. Jhanjhi, and A. Abdullah, "Proposing a privacy protection model in case of civilian drone." pp. 596-602.
- [18] J. I. R. Molano, J. M. C. Lovelle, C. E. Montenegro, J. J. R. Granados, and R. G. Crespo, "Metamodel for integration of internet of things, social networks, the cloud and industry 4.0," *Journal of ambient intelligence and humanized computing*, vol. 9, no. 3, pp. 709-723, 2018.
- [19] M. Al-Emran, S. I. Malik, and M. N. Al-Kabi, "A

- Survey of Internet of Things (IoT) in Education: Opportunities and Challenges," *Toward Social Internet of Things (SIoT): Enabling Technologies, Architectures and Applications*, pp. 197-209: Springer, 2020.
- [20] N. Cvar, J. Trilar, A. Kos, M. Volk, and E. Stojmenova Duh, "The Use of IoT Technology in Smart Cities and Smart Villages: Similarities, Differences, and Future Prospects," *Sensors*, vol. 20, no. 14, pp. 3897, 2020.
- [21] V. Singhal, S. Jain, D. Anand, A. Singh, S. Verma, J. J. Rodrigues, N. Z. Jhanjhi, U. Ghosh, O. Jo, and C. Iwendi, "Artificial Intelligence Enabled Road Vehicle-Train Collision Risk Assessment Framework for Unmanned Railway Level Crossings," *IEEE Access*, vol. 8, pp. 113790-113806, 2020.
- [22] M. Humayun, M. Niazi, N. Jhanjhi, M. Alshayeb, and S. Mahmood, "Cyber Security Threats and Vulnerabilities: A Systematic Mapping Study," *Arabian Journal for Science and Engineering*, pp. 1-19, 2020.
- [23] Z. A. Almusaylim, and N. Zaman, "A review on smart home present state and challenges: linked to context-awareness internet of things (IoT)," *Wireless networks*, vol. 25, no. 6, pp. 3193-3204, 2019.
- [24] W. Trappe, R. Howard, and R. S. Moore, "Low-energy security: Limits and opportunities in the internet of things," *IEEE Security & Privacy*, vol. 13, no. 1, pp. 14-21, 2015.
- [25] J. Saleem, M. Hammoudeh, U. Raza, B. Adebisi, and R. Ande, "IoT standardisation: Challenges, perspectives and solution." pp. 1-9.
- [26] S. S. A. Simnani, S. Shah, and M. T. Banday, "Architectural standards for internet of things: standardising IoT architecture," *International Journal of Forensic Engineering*, vol. 4, no. 3, pp. 196-213, 2019.
- [27] F. Samie, L. Bauer, and J. Henkel, "IoT technologies for embedded computing: A survey." pp. 1-10.
- [28] S. Zeadally, and M. Tsikerdekis, "Securing Internet of Things (IoT) with machine learning," *International Journal of Communication Systems*, vol. 33, no. 1, 2020.
- [29] S. Zeadally, and M. Tsikerdekis, "Securing Internet of Things (IoT) with machine learning," *International Journal of Communication Systems*, vol. 33, no. 1, pp. e4169, 2020.
- [30] F. Meneghello, M. Calore, D. Zucchetto, M. Polese, and A. Zanella, "IoT: Internet of threats? a survey of practical security vulnerabilities in real IoT devices," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8182-8201, 2019.
- [31] Y. Wang, S.-F. Sun, J. Wang, J. K. Liu, and X. Chen, "Achieving Searchable Encryption Scheme with Search Pattern Hidden," *IEEE Transactions on Services Computing*, 2020.
- [32] M. James, and D. S. Kumar, "An implementation of modified lightweight advanced encryption standard in FPGA," *Procedia Technology*, vol. 25, pp. 582-589, 2016.
- [33] F. Maqsood, M. Ahmed, M. Mumtaz, and M. Ali, "Cryptography: A Comparative Analysis for Modern Techniques," *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 6, 2017.
- [34] J. J. Tay, M. M. Wong, and I. Hijazin, "Compact and low power aes block cipher using lightweight key expansion mechanism and optimal number of s-boxes." pp. 108-114.
- [35] T. Sharma, "Lightweight Encryption Algorithms, Technologies, and Architectures in Internet of Things: A Survey," *Innovations in Computer Science and Engineering*, pp. 341-351: Springer, 2020.
- [36] M. El-Haii, M. Chamoun, A. Fadlallah, and A. Serhrouchni, "Analysis of cryptographic algorithms on IoT hardware platforms." pp. 1-5.
- [37] A. Hameed, and A. Alomary, "Security Issues in IoT: A Survey." pp. 1-5.
- [38] I. K. Dutta, B. Ghosh, and M. Bayoumi, "Lightweight Cryptography for Internet of Insecure Things: A Survey." pp. 0475-0481.
- [39] N. Kumar, and N. Reddy, "ESTIMATION OF CRYPTOGRAPHIC APPROACH ON IoT DEVICES," 2019.
- [40] S. Maitra, D. Richards, A. Abdelgawad, and K. Yelamarthi, "Performance Evaluation of IoT Encryption Algorithms: Memory, Timing, and Energy." pp. 1-6.
- [41] M. R. Warner, C. Gardner, R. Wyden, and S. Daines, "Internet of Things Cybersecurity Improvement Act of 2017." p. 1691.
- [42] I. S. Foundation, "IoT Security Foundation," *IoT Security Foundation 2015 - 2020 website: <https://www.iotsecurityfoundation.org/about-us/>*.
- [43] I. S. Foundation, "IOTSF Best Practice Guide," *IoT Security Foundation 2015 - 2020*.
- [44] "Infocom Media Development Authority IMDA "

*2020, Infocomm Media Development Authority.
Last updated 22 May 2020m website:
<https://www.imda.gov.sg/Who-We-Are>.*

- [45] "Infocomm Media Development Authority IMDA Guidelines," 2020, *Infocomm Media Development Authority. Last updated 13 Mar 2020.*



Matasem Saleh is a Ph.D. scholar at Taylor's University, Malaysia. There, he is working as a researcher in the area of Drone Detection System, Privacy Protection and IoT security. He has worked in industry for a decade as Telecom Project Manager in EmaarAltal, Saudi Arabia. Obtained his MSc in Computer Engineering from the University of Engineering and Technology, Pakistan in 2008, where he developed FlocARe, an open source network management software.



Dr. Noor Zaman received his Ph.D. degree in IT from UTP, Malaysia. He has great international exposure in academia, research, administration, and academic quality accreditation. He was with ILMA University, King Faisal University (KFU) for a decade. He is currently with Taylors University, Malaysia. He has 19 years of teaching and administrative experience. He has an intensive background of academic quality accreditation in higher education besides scientific research activities. He had worked a decade for academic accreditation. He received ABET accreditation twice for three programs from CCSIT, King Faisal University, Saudi Arabia. He also worked for National Commission for Academic Accreditation and Assessment (NCAAA), Education Evaluation Commission Higher Education Sector (EECHES) formerly NCAAA Saudi Arabia, for institutional level accreditation. He also worked for National Computing Education Accreditation Council (NCEAC). Dr. Noor Zaman has been awarded the top Reviewer 1% globally by WoS/ISI (Publons) recently. He has edited/authored more than 11 research books with international reputed publishers, received several research grants, and a great number of indexed research articles on his credit. He has supervised several postgraduate students, including master's and Ph.D. He is an Associate Editor of IEEE ACCESS, Keynote speaker for several IEEE international conferences globally, External examiner/evaluator for Ph.D. and master's for several universities, Guest editor of several reputed journals, a member of the editorial board of several research journals, and an active TPC member of reputed conferences around the globe.



Dr. Azween Abdullah is currently working with Taylor's University. He is currently a Professional Development Alumni of Stanford University and MIT. His work experience includes 30 years as an academic in institutions of higher learning and as the Director of research and academic affairs at two institutions of higher learning, the Vice-President for educational consultancy services, 15 years in commercial companies as a Software Engineer, a Systems Analyst, and as a Computer Software Developer and a IT/MIS consultancy and training.



Raazia Saher is a lecturer at King Faisal University in the Department of Computer Science and Information Technology. She has over 10 years of academic experience in this prestigious institution. She has a master's degree in electrical engineering and is a registered professional engineer in Pakistan Engineering Council. She acquired specialized skills in Next Generations Networks & Soft Switches while working as an operational engineer in Pakistan Telecommunications Company Limited.