ISCTE IUL
REPOSITÓRIO
INSTITUTO UNIVERSITÁRIO DE LISBOA

Repositório ISCTE-IUL

Deposited in *Repositório ISCTE-IUL*:
2026-04-01

Deposited version:
Accepted Version

Peer-review status of attached file:
Peer-reviewed

Citation for published item:
Costa, P., Noetzold, D., García Ovejero, R., Martín Esteban, R. & Leithardt, V. (2025). Prisec II: A comprehensive model for IoT security. In Daniel H. de la Iglesia, Juan F. de Paz Santana, Alfonso J. López Rivero (Ed.), Proceedings of the 5th Int. Conference on Disruptive Technologies, Tech Ethics and Artificial Intelligence (DiTTEt 2025). (pp. 147-158). Salamanca: Springer.

Further information on publisher's website:
10.1007/978-3-031-99474-6_13

Publisher's copyright statement:
This is the peer reviewed version of the following article: Costa, P., Noetzold, D., García Ovejero, R., Martín Esteban, R. & Leithardt, V. (2025). Prisec II: A comprehensive model for IoT security. In Daniel H. de la Iglesia, Juan F. de Paz Santana, Alfonso J. López Rivero (Ed.), Proceedings of the 5th Int. Conference on Disruptive Technologies, Tech Ethics and Artificial Intelligence (DiTTEt 2025). (pp. 147-158). Salamanca: Springer., which has been published in final form at https://dx.doi.org/10.1007/978-3-031-99474-6_13. This article may be used for non-commercial purposes in accordance with the Publisher's Terms and Conditions for self-archiving.

Use policy

Creative Commons CC BY 4.0
The full-text may be used and/or reproduced, and given to third parties in any format or medium, without prior permission or charge, for personal research or study, educational, or not-for-profit purposes provided that:

- a full bibliographic reference is made to the original source
- a link is made to the metadata record in the Repository
- the full-text is not changed in any way

The full-text must not be sold in any format or medium without the formal permission of the copyright holders.

Serviços de Informação e Documentação, Instituto Universitário de Lisboa (ISCTE-IUL)
Av. das Forças Armadas, Edifício II, 1649-026 Lisboa Portugal
Phone: +(351) 217 903 024 | e-mail: administrador.repositorio@iscte-iul.pt
https://repositorio.iscte-iul.pt

Prisec II: A Comprehensive Model for IoT Security

Pedro da Costa[1], Darlan Noetzold[2], Raúl García Ovejero[3], Roberto Martín Esteban[3,4], and Valderi R. Q. Leithardt[5]

[1] Instituto Superior de Engenharia de Lisboa (ISEL), Lisboa, Portugal
a49944@alunos.isel.pt
[2] University of Vale do Rio dos Sinos (UNISINOS), São Leopoldo, Brazil
dnoetzold@edu.unisinos.br
[3] Expert Systems and Applications Laboratory, E.T.S.I.I. of Béjar, Universidad de Salamanca, 37700 Salamanca, Spain
raulovej@usal.es
[4] IBERDROLA, Research power generation, Spain
rmartines@usal.es
[5] Instituto Universitário de Lisboa (ISCTE-IUL), ISTAR, Av. das Forças Armadas, 1649-026, Lisboa, Portugal
valderi.leithardt@iscte-iul.pt

Abstract. This study examines data security and efficiency in interconnected IoT devices, focusing on selecting cryptographic algorithms to improve secure data transmission in 5G networks. The proposed model introduces four security levels, each applying different encryption strategies to balance security and performance. Cloud computing is integrated to mitigate computational limitations in IoT devices, optimizing encryption and decryption processes. The study evaluates multiple cryptographic algorithms, analyzing encryption and decryption times, packet throughput, and memory usage. Experimental results demonstrate that the model enables efficient encryption while maintaining security, with AES-256 and XChaCha20 showing stable performance across different packet sizes. The cloud-based implementation improves resource distribution and reduces processing delays. This work contributes a structured cryptographic model adaptable to varying security needs and a performance evaluation of different cryptographic approaches in cloud-integrated environments.

Keywords: Algorithms, Cryptographic, Cloud Computing Optimization, Data Transmission Efficiency, Prisec

1 Introduction

With the increasing prevalence of cloud computing, Edge and Big Data in both everyday life and business operations, ensuring security of data processed by IoT devices against potential intruders has become a paramount concern [26]. The fields of medicine, smart grids, home automation, precision agriculture and urban mobility represent just a few of the many scenarios where cloud computing

is present [5]. However, with the benefits of connectivity also come significant risks, particularly regarding data security, which is the focus of this study. In this context, cryptography plays a vital role in safeguarding sensitive information since this technique consists of developing and using coded algorithms to protect and obscure transmitted information so that it may only be read by those with the permission and ability to decrypt it [16]. In summary, cryptography plays a significant role in data security ensuring that the data transmitted amidst interconnected system stays protected. Cloud computing, on the other hand, refers to the delivery of computing services including storage, databases, networking, and other resources over the internet. It offers numerous benefits, such as security, speed, cost, and others. There are numerous uses for cloud computing such as testing and building applications, streaming audio and video, analyzing data, and others [15]. In this study, the cloud will be used for testing and build applications together with some cryptography algorithms, being this explained further in this paper.

Nowadays, IoT applications are structured into four layers: sensing, network, middleware, and application. Each layer employs different technologies, introducing various security risks [23]. The application layer interacts directly with users and faces unique threats like data theft and DDoS attacks [25]. Due to the high volume of sensitive data in motion, IoT applications are more vulnerable to attacks than data at rest, making encryption a key security measure [15]. This study addresses these security concerns by implementing cryptographic techniques, focusing on algorithms suited for 5G networks. The goal is to evaluate encryption methods based on packet throughput and size to enhance IoT security and protect sensitive data from unauthorized access and manipulation.

The main contribution of this work is the design and implementation of Prisec II, a novel cryptographic framework that introduces a multi-layered, cloud-integrated approach to secure data transmission in IoT environments. Unlike existing models, Prisec II combines adaptive encryption strategies with a hierarchical access model and resource-aware execution, enabling effective performance on constrained edge devices. This dual focus on security robustness and system efficiency represents a distinct advancement over traditional encryption frameworks limited to single-layer architectures or fixed cryptographic protocols.

The first goal is to select and analyze different cryptographic algorithms to identify the algorithms with the best performance in terms of encryption and decryption time, while considering the size of the packet. The next objective is to implement these algorithms in the structure model that will be further explained in the report to evaluate the time taken to encrypt and decrypt different sizes of packets. After finding the best structure model, the next objective is to analyze the performance of this model regarding the number of packets transmitted, after that this structure model will be used together with cloud computing while also evaluating the performance of the model.

2 Related Works

It is common knowledge that security is an increasing concern in real-time systems, so researchers have demonstrated attacks and defenses aimed at such systems. In the article [12], it was identified, classified, and measured the effectiveness of the security research in this domain. It provided a high-level summary [identification] and a taxonomy [classification] of this existing body of work. Furthermore, it carried out an in-depth analysis [measurement] of scheduler-based security techniques — the most common class of real-time security mechanisms. The comparative analysis of these techniques is summarized in Table 1. For this purpose, it developed a common metric, "attacker's burden," used to measure the effectiveness of scheduler-based real-time security measures.

Decentralized systems distribute computational tasks across multiple nodes without a central authority, making security a significant challenge, especially when nodes have varying resource capacities. Prior works such as [6, 24] evaluated security solutions and benchmarked cryptographic algorithms under different hardware constraints to identify the necessary computing capabilities for secure operations.

In response to growing concerns over data breaches and unauthorized access in networked systems, various cryptographic models have been proposed. These include approaches combining encryption, hash functions, and digital signatures to ensure data integrity and secure communication. For example, [4] introduced a framework using ECC and AES for wireless sensor networks, while [13] analyzed the performance of symmetric ciphers like AES, RC6, and ChaCha20-Poly1305. Additionally, [14] assessed the impact of cryptographic algorithms on resource-constrained smart home devices, examining power consumption, delay, and message overhead.

Further comparative analyses of symmetric block-based cryptographic algorithms, including AES, CLEFIA, DES, Triple DES, IDEA, PRESENT, SEA, SPECK, TEA, XTEA, and TWOFISH, provided insights into energy efficiency, power consumption, and memory usage [8]. Another study [2] examined AES, DES, RSA, and the lightweight cryptographic algorithm Fernet to identify the most suitable option for minimizing security risks in IoT applications. The feasibility of lightweight cryptography for resource-constrained smart devices was explored in [7], evaluating execution time, code efficiency, and communication overhead. Table 1 compares these studies with the proposed model, considering aspects like packet size variation, algorithm diversity, encryption and decryption times, cloud computing integration, CPU impact, and memory usage. Additionally, research in [17–22] discusses encryption strategies and data management in various application scenarios, contributing to a broader understanding of cryptographic solutions in IoT environments.

Building upon insights from these studies, our work aims to develop a robust cryptographic model that differs from previous models by employing encapsulation to enhance data security. Table 1 presents a comparative analysis between the cited works and our proposed model, highlighting key aspects. The Size Var. column indicates whether different packet sizes (e.g., KB, MB) were considered

in the respective studies. The Alg. Var. column shows whether multiple cryptographic algorithms were analyzed. The Time column assesses if encryption and decryption times were used for performance comparison. The Cloud column identifies whether the study incorporated cloud computing for encryption or storage processes. The CPU column specifies whether the study examined the impact of computational load on processing units. Finally, the Mem. column evaluates whether memory usage was considered in the cryptographic implementation. Each column helps illustrate the scope and depth of the security evaluations performed in the analyzed studies.

Table 1. Comparative Analysis

Work	Size Var.	Alg. Var.	Time	Cloud	CPU	Mem.
[1]		X				
[2]		X	X			
[4]		X				
[7]		X	X			
[8]		X	X			X
[13]		X	X		X	X
[14]		X	X			
Prisec II	X	X	X	X	X	X

Unlike the cited models, the proposed model integrates multiple cryptographic algorithms in layers, preventing attackers from easily identifying encryption patterns. For example, if plaintext is encrypted with AES, the ciphertext is further encrypted using another algorithm. Additionally, this study integrates cloud computing to address computational limitations inherent to IoT devices.

While existing studies provided valuable insights into the performance of cryptographic algorithms, the design of the Prisec II model required addressing several challenges not fully explored in prior work. Key design decisions included selecting algorithms capable of maintaining performance across a range of packet sizes, defining hierarchical security levels suited to different application contexts, and implementing a layered encryption strategy to prevent pattern detection. Additionally, integration with cloud environments required careful consideration of resource allocation and latency management. These choices were guided by gaps identified in related studies, as summarized in Table 1, positioning Prisec II as a differentiated and adaptable solution for secure IoT communication.

3 Methodology

This study enhances data security and performance in interconnected IoT devices by selecting and implementing cryptographic algorithms while leveraging cloud computing for optimization. A literature review guided the selection of algorithms, focusing on those applicable to 5G networks and integrating 4G network algorithms for broader relevance. The evaluation considered encryption and

decryption times, packet size handling, and packet throughput. Figure 1 illustrates the system architecture used in these assessments. The primary algorithms analyzed include AES [9] (modes CBC and CTR), Blowfish [9], Chacha20 and XChacha20 [10], [11] (Elliptic Curve Cryptography Diffie–Hellman), and HMAC-SHA-256 [3] (Hash-based Message Authentication Code using SHA-256).

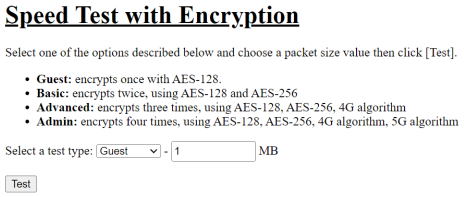


Fig. 1. First prototype and interface created.

The implementation phase focused on developing a Python-based program using the PyCryptodome library to evaluate the performance of various cryptographic algorithms. A multilayered encryption model was designed with four security levels: Guard, Basic, Advanced and Admin, where higher levels involve more complex encryption processes, such as triple encryption at the Advanced level, increasing security but also computational overhead. Performance tests were conducted on machines with AMD Ryzen processors and different RAM capacities to determine the optimal encryption and decryption configurations. Initial tests assessed encryption and decryption times for different packet sizes and algorithm combinations, as well as the impact of varying packet quantities.

The next phase integrated cloud computing to reduce the computational limitations of IoT devices by offloading cryptographic operations. Figure 2 illustrates the system architecture, where the encryption and decryption processes were separated and executed in the cloud to improve efficiency. In addition, a new interface was created to enhance the usability and management of the cryptosystem, as shown in Figure 3. The performance of the structured model in a cloud environment was analyzed using Microsoft Azure and a Raspberry Pi 4 Model B as the primary tools for implementation. The Azure virtual machine had 8 GB of RAM and an Intel-based processor, while the Raspberry Pi featured 8 GB of RAM and a Broadcom BCM2711 quad-core Cortex-A72 processor.

This analysis focused on measuring key metrics such as encryption and decryption times, the impact of packet size, packet throughput, memory usage at each security level, and client-server processing times. The system was designed with the virtual machine acting as the client for the decryption process, while the Raspberry Pi managed encryption and transmitted data to the cloud-based virtual machine.

Through analysis of previous stages, it was determined that a system where encryption and decryption are performed on separate machines requires at least

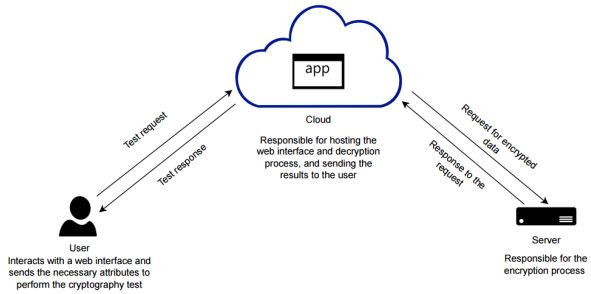


Fig. 2. Cryptosystem created.

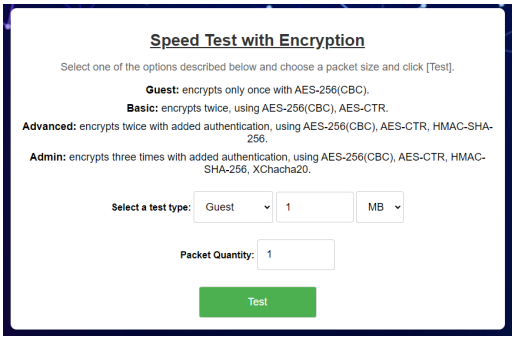


Fig. 3. New interface implemented.

16 GB of RAM on each to match the packet throughput observed when both processes were executed on a single 32 GB machine.

4 Results

The initial testing phase evaluated encryption and decryption times for different packet sizes using ChaCha20(4G) and Blowfish(5G) in the structure model. The Guest and Admin levels were selected for comparison, representing the simplest and most complex configurations, respectively. Packet sizes tested included 1, 5, 10, 20, and 50 MB, confirming that processing time increases with packet size. The Guest level showed the best performance with minimal processing time, whereas the Admin level had longer encryption and decryption times.

The next phase replaced Blowfish or ChaCha20 with AES-CTR to improve performance while maintaining security. For smaller packets, ChaCha20 and AES-CTR showed similar performance to Blowfish and AES-CTR, but for larger packets, ChaCha20 performed better. At 50 MB, ChaCha20 took 265 milliseconds for encryption and 290 milliseconds for decryption, whereas Blowfish took 397 milliseconds and 407 milliseconds, respectively. The comparison revealed

that AES-CTR offered a balanced approach by reducing processing time while maintaining high encryption security.

A new structure model incorporated AES-CTR with ECDH and HMAC-SHA-256 at the Admin level, each introduced separately. Results showed that while ECDH enhanced security, it increased processing time. For smaller packets (1 MB and 5 MB), HMAC-SHA-256 performed better, but as packet size increased, ECDH was more efficient. At 50 MB, ECDH's encryption time was 0.9 seconds, compared to 1.01 seconds with HMAC-SHA-256. This demonstrates that while hash-based authentication methods may be suitable for smaller data transmissions, key exchange mechanisms like ECDH are better suited for handling larger packets securely.

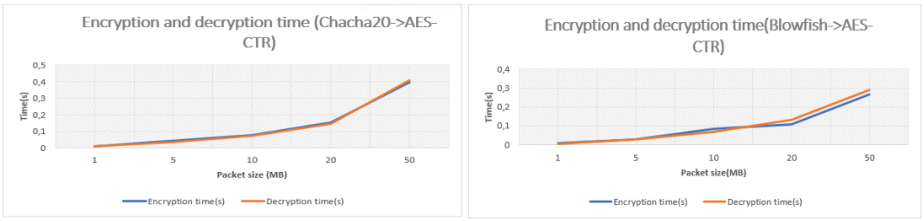


Fig. 4. Performance of Chacha20 and Blowfish with AES-CTR.

AES-256 consistently outperformed other algorithms across all packet sizes, achieving an encryption time of 0.0013 seconds for 1 MB and 0.0219 seconds for 50 MB, significantly surpassing ECC (0.08 seconds) and Blowfish (0.31 seconds). The next evaluation phase analyzed the impact of packet throughput using the optimal encryption model on machines with 16 GB and 32 GB of RAM. The Admin level was used to assess computational limitations. Figure 5 presents the encryption and decryption times for varying packet quantities and sizes.

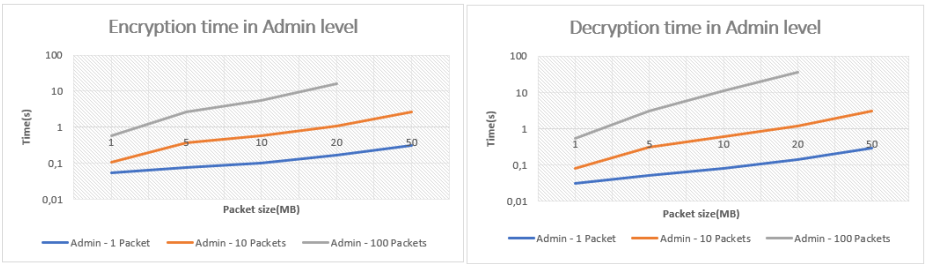


Fig. 5. Impact of packet throughput.

Encryption times increased nearly linearly with packet size and quantity. For 1, 10, and 100 packets, larger packet sizes resulted in longer encryption

times, with 100-packet transmissions taking the most time. Decryption followed a similar trend. With 16 GB RAM, the system could handle up to 10 packets of 50 MB and 100 packets of 20 MB before performance degradation. A second test on a machine with 32 GB RAM confirmed performance improvements, enabling transmission of up to 10 packets of 50 MB and 1000 packets of 20 MB. To further address computational constraints, a new test introduced smaller packet sizes (1 KB to 120 KB) on the 32 GB machine. As shown in Figure 6, reducing packet size significantly increased the number of packets that could be processed simultaneously.

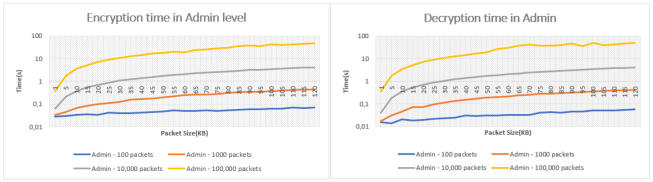


Fig. 6. Impact of packet throughput on KB packets.

The final phase evaluated the cryptosystem, deploying a virtual machine on Microsoft Azure (8 GB RAM) for decryption, while a Raspberry Pi acted as the encryption server. Due to compatibility issues, ECDH was replaced with XChaCha20. The client-side application, built with Flask, managed encryption and decryption requests, ensuring secure and efficient communication between the encryption and decryption endpoints. Figure 7 presents the encryption and decryption times for different packet quantities, ranging from 1 to 30 packets, at the Admin level. The results demonstrate a clear trend: as the packet size and number of packets increase, both encryption and decryption times rise. This trend is expected due to the higher computational requirements associated with processing larger data volumes. In the encryption graph, a near-exponential growth in processing time is observed, particularly when moving from lower to higher packet sizes.

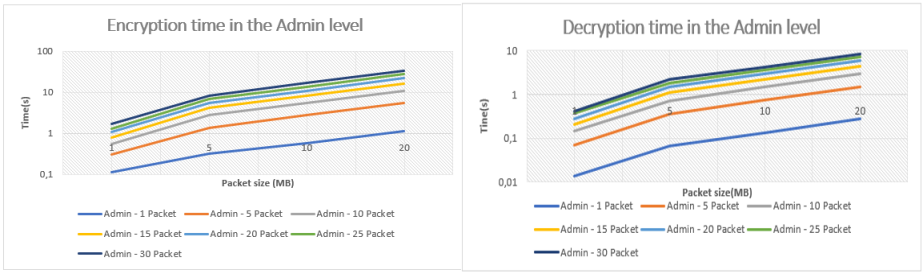


Fig. 7. Performance of encryption and decryption in the cryptosystem.

The impact of CPU architecture was evident, as the Intel-based cloud machine performed encryption and decryption faster than the ARM-based Raspberry Pi. Memory usage analysis (Figure 8) revealed that the Admin level consumed the most memory, while the Guest level used the least.

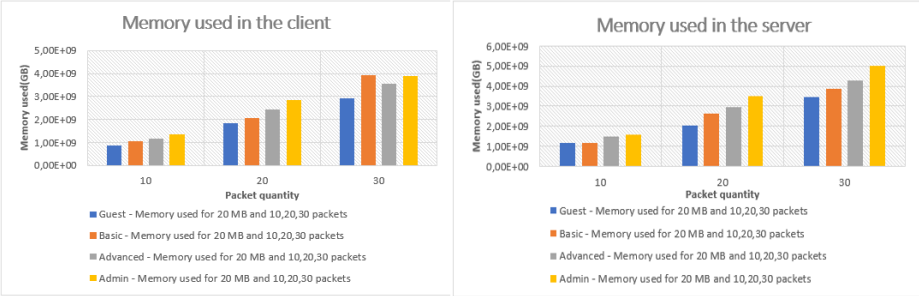


Fig. 8. Memory usage in the cryptosystem.

This study highlights the trade-offs between security and performance. Larger packets and higher encryption levels increase processing times, but machines with more RAM mitigate this impact. The results emphasize the importance of selecting appropriate cryptographic algorithms and optimizing computational resources for secure and efficient IoT applications.

5 Conclusion

In this study, we addressed the pressing issue of ensuring data security in interconnected IoT devices, particularly within the context of 5G networks. Our approach involved a comprehensive analysis and selection of efficient cryptographic algorithms to design a model that optimizes data transmission while maintaining strong security standards. A significant part of our solution was the integration of cloud computing, which enhanced processing efficiency and resource utilization. This integration not only improved overall security but also boosted transmission performance by allowing the offloading of complex cryptographic computations to the cloud, thus reducing the processing burden on individual IoT devices.

The scientific contribution of this work is the creation of a hierarchical security model structured into four levels: Guest, Basic, Advanced, and Admin. Each level, except Guest, incorporates encapsulation with specific cryptographic algorithms tailored to the sensitivity of data and user privileges. By leveraging cloud computing, we enhanced the processing capabilities and resource efficiency of the model, which is crucial for ensuring scalability and robustness in practical applications. Offloading cryptographic tasks to the cloud allowed us to reduce computational overhead, a critical factor for the practical deployment of secure IoT networks without compromising on performance.

The best-performing structured model combined AES-256, AES-CTR, HMAC-SHA-256, and XChaCha20, demonstrating strong performance across various tests. This approach enabled the transmission of 1000 packets of 20 MB on a single machine with 32 GB of RAM for both encryption and decryption. Distributing these processes between two machines would reduce the memory requirement to 16 GB per machine, emphasizing the role of efficient memory management in optimizing performance and cost. Extensive testing, graphical analysis, and algorithm adjustments were conducted to identify the optimal structure model.

The proposed cryptographic model is applicable to real-world scenarios such as smart city infrastructures, healthcare monitoring systems, and industrial IoT applications, where layered security and computational efficiency are important. Future deployment may benefit from alignment with security standards such as NIST SP 800-183 for IoT device security and interoperability frameworks like oneM2M or the IEEE P2413 IoT architecture. These integrations can support adoption in heterogeneous environments while addressing industry requirements for resilience and trust.

Looking ahead, there are areas for further exploration. Future work could focus on optimizing cryptographic algorithms to reduce computational overhead and improve efficiency. Testing the scalability of the proposed model in larger, real-world IoT networks is also important to ensure robustness. Additionally, exploring the use of hybrid encryption techniques that combine symmetric and asymmetric algorithms could enhance security and performance. Integrating emerging technologies such as edge computing and blockchain with our model presents another promising avenue for improving data security. Furthermore, applying the structured model to UBiPri hardware [27], which was previously used in a 4G network cryptographic project, would allow for a comparison of advancements from 4G to 5G. Incorporating cryptographic algorithms suitable for 6G networks and adding HTTPS to the communication between client and server are additional security measures to consider. Finally, further optimization of memory usage at each level of the security model will help balance resource costs while maintaining performance.

Acknowledgments

A realização desta investigação foi parcialmente financiada por fundos nacionais através da FCT - Fundação para a Ciência e Tecnologia, I.P. no âmbito dos projetos UIDB/04466/2020 e UIDP/04466/2020. A realização desta investigação foi parcialmente financiada por fundos nacionais através da FCT - Fundação para a Ciência e Tecnologia , I.P. no âmbito do projeto 16881, LISBOA2030-FEDER-00816400, (In Portuguese).

References

1. Jain, A., Gupta, H.: A Cryptographic Model for the Enhancement of Data Security. In: 2024 11th International Conference on Reliability, Infocom Technologies

and Optimization (Trends and Future Directions) (ICRITO), Noida, India, pp. 1-6 (2024). DOI: 10.1109/ICRITO61523.2024.10522303

2. David, V., Ragu, H., Nikhil, V., Sasikumar, P.: Analysis of Algorithms for Effective Cryptography for Enhancement of IoT Security. In: Zhang, Y. D., Senjyu, T., So-In, C., Joshi, A. (eds.) Smart Trends in Computing and Communications. Lecture Notes in Networks and Systems, vol. 396, Springer, Singapore (2023). DOI: 10.1007/978-981-16-9967-2_10
3. Ravilla, D., Putta, C. S. R.: Implementation of HMAC-SHA256 algorithm for hybrid routing protocols in MANETs. In: 2015 International Conference on Electronic Design, Computer Networks & Automated Verification (EDCAV), Shillong, India, pp. 154-159 (2015). DOI: 10.1109/EDCAV.2015.7060558
4. Dayana, D. S., Pandian, R., Babu, A. R., Nirmalraj, S., Jebaseelan, S. D. S., M. V.: Elevating Security in Wireless Sensor Networks using ECC and AES Cryptographic Techniques. In: 2023 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES), Chennai, India, pp. 1-6 (2023). DOI: 10.1109/ICSES60034.2023.10465296
5. De Moura Costa, H. J., Da Costa, C. A., Da Rosa Righi, R., Antunes, R. S., De Paz Santana, J. F., Leithardt, V. R. Q.: A Fog and Blockchain Software Architecture for a Global Scale Vaccination Strategy. IEEE Access, vol. 10, pp. 44290-44304 (2022). DOI: 10.1109/ACCESS.2022.3169418
6. Hu, Z.: Cryptography, Dependability and Privacy in Decentralized Systems. Ph.D. thesis, University of Oregon (2023). Accessed: January 2025. Available: https://scholarsbank.uoregon.edu/xmlui/handle/1794/29133
7. Lino, I., Cecílio, J.: A Comparative Analysis of the Impact of Cryptography in IoT LoRa Applications. In: 2022 IEEE 20th International Conference on Industrial Informatics (INDIN), Perth, Australia, pp. 220-225 (2022). DOI: 10.1109/INDIN51773.2022.9976108
8. Makarenko, I., Semushin, S., Suhai, S., Kazmi, S. M. A., Oracevic, A., Hussain, R.: A Comparative Analysis of Cryptographic Algorithms in the Internet of Things. In: 2020 Inter. Sci. and Technical Conference Modern Computer Network Technologies (MoNeTeC), (2020). DOI: 10.1109/MoNeTeC49726.2020.9258156
9. Gaur, J. D., Kumar Singh, A., Singh, N. P., Rajan, G. V.: Comparative Study on Different Encryption and Decryption Algorithm. In: 2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), Greater Noida, India, pp. 903-908 (2021). DOI: 10.1109/ICACITE51222.2021.9404734
10. Kebande, V.: Extended-Chacha20 Stream Cipher With Enhanced Quarter Round Function. IEEE Access, (2023). DOI: 10.1109/ACCESS.2023.3324612
11. Koppl, M. et al.: Performance Comparison of ECDH and ECDSA. In: 2021 2nd International Conference on Electronics, Communications and Information Technology (CECIT), Sanya, China, pp. 825-829 (2021). DOI: 10.1109/CECIT53797.2021.00149
12. Hasan, M., Kashinath, A., Chen, C., Mohan, S.: SoK: Security in Real-Time Systems. ACM Comput. Surv., (2024). DOI: 10.1145/3649499
13. Saraiva, D. A. F., Leithardt, V. R. Q., de Paula, D., Mendes, A. S., González, G. V., Crocker, P.: PRISEC: Comparison of Symmetric Key Algorithms for IoT Devices. Sensors, vol. 19, no. 4312 (2019). DOI: 10.3390/s19194312

14. Silva, C., Cunha, V. A., Barraca, J. P. et al.: Analysis of the Cryptographic Algorithms in IoT Communications. Inf Syst Front (2023). DOI: 10.1007/s10796-023-10383-9
15. Hassija, V., Chamola, V., Saxena, V., Jain, D., Goyal, P., Sikdar, B.: A Survey on IoT Security: Application Areas, Security Threats, and Solution Architectures. IEEE Access, vol. 7, pp. 82721-82743 (2019). DOI: 10.1109/ACCESS.2019.2924045
16. Microsoft Azure: What is Cloud Computing? Accessed: January 2025. Available at: https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-is-cloud-computing
17. Pereira, F., Crocker, P., Leithardt, V. R. Q.: PADRES: Tool for PrivAcy, Data REgulation and Security. SoftwareX, vol. 17, pp. 100895 (2022). DOI: 10.1016/j.softx.2021.100895
18. Morais, R., Crocker, P., Leithardt, V. R. Q.: Nero: A Deterministic Leaderless Consensus Algorithm for DAG-Based Cryptocurrencies. Algorithms, vol. 16, no. 1, pp. 38 (2023). DOI: 10.3390/a16010038
19. Pereira, M. M., Leithardt, V. R. Q.: Fake News Management: An Approach Using Collaborative Networks. Accessed: January 2025. Available at: https://ssrn.com/abstract=4807081.
20. Leithardt, V. R. Q. et al.: A Privacy Taxonomy for the Management of Ubiquitous Environments. J. Commun. Comput., vol. 10, no. 17265, pp. 1548-7709 (2013). Accessed: January 2025. Available at: https://www.davidpublisher.com/Public/uploads/Contribute/5539ed57ad7f0.pdf
21. Leithardt, V. R. Q. et al.: Privacy Management Solution in Ubiquitous Environments Using Percontrol. J. Ubiquitous Syst. Pervasive Netw., vol. 5, no. 2, pp. 21-28 (2014). Accessed: January 2025. Available at: https://iasks.org/articles/juspn-v05-i2-pp-21-28.pdf
22. Ochoa, I. S., Leithardt, V. R. Q., Zeferino, C. A., Silva, J. S.: Data Transmission Performance Analysis with Smart Grid Protocol and Cryptography Algorithms. In: 2018 13th IEEE International Conference on Industry Applications (INDUSCON), São Paulo, Brazil, (2018). DOI: 10.1109/INDUSCON.2018.8627195
23. Noetzold, D., Rossetto, A. G. d. M., Silva, L. A., Crocker, P., Leithardt, V. R. Q.: JVM Optimization: An Empirical Analysis of JVM Configurations for Enhanced Web Application Performance. SoftwareX, vol. 28, pp. 101933 (2024). DOI: 10.1016/j.softx.2024.101933
24. Noetzold, D., Rossetto, A. G. d. M., Izquierdo, L. R., Leithardt, V. R. Q.: Use of Spyware Integrated with Prediction Models for Computer Monitoring. In: 2023 18th Iberian Conference on Information Systems and Technologies (CISTI), Aveiro, Portugal, pp. 1-6 (2023). DOI: 10.23919/CISTI58278.2023.10211594
25. Jaafar, G. A., Abdullah, S. M., Ismail, S.: Review of Recent Detection Methods for HTTP DDoS Attack. J. Comput. Netw. Commun., vol. 2019, no. 1, pp. 1283472 (2019). DOI: 10.1155/2019/1283472
26. Garg, S., Kaur, K., Kaddoum, G., Garigipati, P., Aujla, G. S.: Security in IoT-Driven Mobile Edge Computing: New Paradigms, Challenges, and Opportunities. IEEE Network, (2021). DOI: 10.1109/MNET.211.2000526
27. Lopes, Hugo and Pires, Ivan Miguel and Sánchez San Blas, Hector and García-Ovejero, Raúl and Leithardt, Valderi.: PriADA: Management and Adaptation of Information Based on Data Privacy in Public Environments. Journal Computers, (2020). DOI: 10.3390/computers9040077