LIGHTWEIGHT SYMMETRIC KEY PROTOCOL WITH SECURE COMMUNICATION FOR RESOURCE-CONSTRAINED IOT DEVICES

Samsheer P Irfan Alikhan[1], Md. Ateeq Ur Rahman[2], Subramanian K.M[3]

[1]PG Scholar, Department of Computer Science and Engineering,
Shadan College of Engineering and Technology, Hyderabad, Telangana, India – 500086
Email: irfanalikhan423@gmail.com
[2]Professor, Department of Computer Science and Engineering,
Shadan College of Engineering and Technology, Hyderabad, Telangana, India – 500086
Email: mail_to_ateeq@yahoo.com
[3]Professor, Department of Computer Science and Engineering,
Shadan College of Engineering and Technology, Hyderabad, Telangana, India – 500086
Email: kmsubbu.phd@gmail.com

Abstract: The Internet of Things (IoT) significantly enhances the convenience of people's daily lives, but a lack of security practices increases the risk of sensitive user data leakage. In IoT environments such as Intelligent Connected Vehicles, Smart Homes, Intelligent Cities, etc., securing the transmission of data between IoT devices is a crucial capability. However, the limited resources of low-cost IoT devices pose a challenge for cryptographic communication schemes; even negligible additional CPU utilization of battery-powered sensors would result in a dramatic decrease in battery life. In this work, to minimize the resource consumption, we propose a communication protocol involving only the symmetric key-based scheme, which provides ultra-lightweight yet effective encryptions to protect the data transmissions. Symmetric keys generated in this protocol are delegated based on a chaotic system, i.e., Logistic Map, to resist against the key reset and device capture attacks. We semantically model such protocol and analyze the security properties. Moreover, the resource consumption is also evaluated to guarantee runtime efficacy.

"Index Terms: *Internet of Things (IoT), symmetric key cryptography, lightweight security protocol, Logistic Map, chaotic encryption, resource-constrained devices".*

1. INTRODUCTION

IoT systems have become an essential component of modern digital infrastructure, enabling pervasive connectivity across smart homes, smart cities, Intelligent Connected Vehicles, and other emerging application domains [1][2][3]. These environments support real-time monitoring, automation, and intelligent decision-making by interconnecting heterogeneous devices that continuously generate and exchange data. The rapid expansion of such ecosystems has significantly transformed industrial operations, healthcare systems, transportation networks, and urban management frameworks. As a result, reliable and secure communication among distributed devices has become a foundational requirement for sustaining trust and functionality in large-scale deployments. Growing interconnectivity also introduces complexity in maintaining consistent trust relationships among devices operating across distributed and often untrusted environments

especially in resource-limited scenarios where traditional security models struggle significantly constrained.

Despite these advancements, ensuring secure data transmission in IoT environments remains a critical challenge due to the exposure of wireless communication channels to potential adversaries [4]. Existing security mechanisms often rely on computationally intensive cryptographic techniques that are not well-suited for resource-constrained devices, leading to increased energy consumption and reduced operational efficiency [5]. Furthermore, many traditional security frameworks fail to address scalability issues and are vulnerable to evolving threats such as unauthorized access, impersonation, and device compromise, which significantly weaken overall system resilience [6]. Additionally, the heterogeneity of IoT ecosystems introduces further difficulties in standardizing security solutions, as devices vary widely in computational capacity, communication protocols, and deployment environments, making uniform protection strategies less effective across diverse application domains in real-world deployments globally systems.

To address these limitations, this work aims to develop a lightweight and secure communication framework tailored for heterogeneous IoT environments, focusing on reducing computational overhead while maintaining strong security guarantees [7]. The approach emphasizes efficient key management and robust authentication mechanisms suitable for large-scale deployments involving diverse device capabilities [8]. Additionally, the framework is designed to enhance resilience against device-level threats and ensure secure communication even in highly dynamic network conditions [9]. Emphasis is also placed on ensuring adaptability to evolving network requirements and maintaining consistent performance under constrained operational conditions in distributed environments while preserving system reliability across heterogeneous deployments efficiently.

Beyond technical improvements, the significance of this effort lies in enabling scalable and energy-efficient secure communication across heterogeneous IoT ecosystems, thereby supporting the next generation of intelligent interconnected systems [10]. Such advancements are expected to contribute to improved trust, reduced energy overhead, and enhanced sustainability in next-generation connected infrastructures operating at scale globally deployed systems.

2. LITERATURE REVIEW

Key management and secure communication in distributed sensor networks have been extensively studied as foundational components of IoT security. Early approaches such as the random key predistribution scheme introduced in [11] established a probabilistic framework for enabling secure links among sensor nodes, improving scalability in large-scale deployments. Similarly, [12] enhanced resilience by enabling random key sharing strategies that increase connectivity probability among devices. The work in [13] provided a theoretical foundation for perfectly secure key distribution in dynamic environments, while [14] introduced efficient pairwise key establishment mechanisms to strengthen secure node-to-node communication. Although these methods significantly contributed to distributed security design, they are often constrained by vulnerability to node capture attacks, limited adaptability to dynamic IoT environments, and

increased storage or communication overhead in dense networks.

Subsequent advancements attempted to address these limitations by improving adaptability and efficiency in heterogeneous deployments. The multi-phase deployment strategy in [15] enhanced flexibility in large-scale sensor networks by supporting staged initialization, while [16] introduced collaborative key management techniques aimed at improving coordination among distributed smart objects. In addition, [17] proposed a location-aware key management mechanism to improve context-based security, and [18] explored algebraic structures such as Kronecker product-based methods for secure key organization in IoT environments. Despite these improvements, these approaches often rely on structural assumptions about network topology or device capabilities, which limits their applicability in highly dynamic and heterogeneous IoT ecosystems. Moreover, many of these schemes still struggle with balancing security strength and computational efficiency under strict resource constraints.

In the context of IoT application-specific security, several protocols have been designed for constrained environments such as smart homes. The privacy-preserving communication mechanism in [19] demonstrates improved data protection for controlled IoT settings with authorized access. However, its applicability remains limited in open and adversarial environments such as Intelligent Connected Vehicles and smart city infrastructures, where devices operate under higher exposure risks and mobility conditions. These limitations highlight a broader gap in existing literature: the lack of a unified security mechanism capable of supporting heterogeneous IoT scenarios while maintaining low computational overhead and robust resistance to evolving attack vectors across diverse deployment environments.

To address these challenges, recent research has explored the use of chaotic systems for cryptographic applications. The generalized bidirectional tent map introduced in [20] demonstrates the potential of chaos-based designs in achieving strong unpredictability for encryption purposes. Such methods offer improved randomness and resistance against cryptanalytic attacks; however, they often lack robust key management strategies and may introduce synchronization challenges in distributed environments. Additionally, their integration into resource-constrained IoT devices remains insufficiently optimized. Motivated by these limitations, the present direction focuses on developing a lightweight symmetric key-based communication framework integrated with chaos-driven key evolution, aiming to enhance security, scalability, and efficiency simultaneously across heterogeneous IoT deployments.

3. MATERIALS AND METHODS

The pervasive integration of IoT devices in smart homes, smart cities, and Intelligent Connected Vehicles (ICVs) introduces a critical problem of compromised data security. Wireless channels commonly used for communication expose privacy-sensitive information to potential eavesdroppers, particularly in outdoor IoT environments vulnerable to attacks. The Electronic Control Units (ECUs) governing ICV logic lack authentication measures, making them susceptible to adversarial manipulations. Current security protocols reliant on resource-intensive asymmetric key-based schemes, such as RSA, DHKE, or ECC, are impractical for low-powered devices. The absence of a tailored, generic

symmetric key-based secure protocol exacerbates the vulnerability of easily-approached IoT systems, necessitating a comprehensive solution to safeguard data integrity and user privacy.

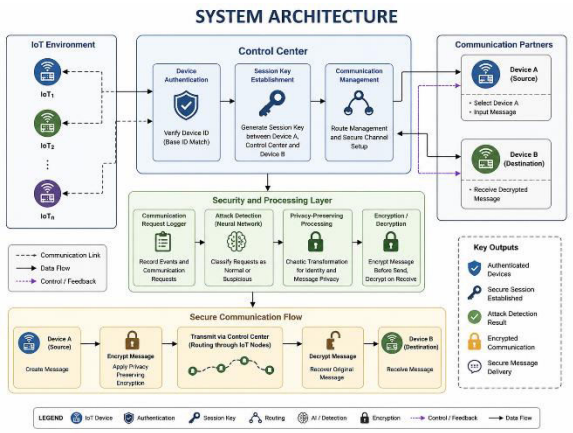


Fig.1 Proposed Architecture

a) Modules:

1) System Initialization and IoT Network Generation: The implementation begins with the creation of a simulated heterogeneous IoT environment through the graphical interface. The system generates one centralized Control Center and multiple IoT devices positioned dynamically within the communication area. During initialization, device coordinates are allocated while maintaining communication spacing constraints. A chaotic value is generated and used to support subsequent identity generation and authentication procedures.
2) Device Identity Generation and Registration: After network generation, unique identities are created for all IoT devices. Each device is assigned a secret identity generated using HMAC-based encoding combined with a generated chaotic parameter. The generated identities are stored internally and used as reference values during authentication. This phase establishes trusted identities before communication begins.
3) Device Authentication and Communication Validation: The communication process starts by selecting Device A and validating its identity through the Control Center. The received device identity is regenerated and verified against the stored identity information. Authentication events are recorded into communication logs representing normal or suspicious requests. This stage determines whether communication requests should proceed further.
4) Attack Monitoring and Request Classification: Communication requests are transformed into structured event records and processed for attack identification. Event information and device-related attributes are prepared into machine-readable format and passed into a pre-trained neural-network-based classification model. The model analyzes communication behavior and categorizes requests as normal communication or suspicious activity before message transfer.
5) Session Establishment and Secure Communication: After successful validation, the system establishes a communication session between Device A, the Control Center, and Device B. Routing is performed through neighboring IoT nodes to simulate practical communication flow. A secure channel is prepared for transferring communication data across the network.
6) Privacy-Preserving Message Protection and Data Exchange: Before transmission, the user-provided message undergoes privacy-preserving processing and encryption. Encrypted communication is transmitted from the source device through the Control Center to the destination device. At the receiver side, the message is decrypted and

reconstructed to complete secure data exchange between selected IoT devices.

7) Communication Execution and Visualization: The implemented interface visualizes communication establishment and message transfer through animated communication paths between participating devices. This execution stage demonstrates authentication, session establishment, secure communication, and final message delivery within the heterogeneous IoT environment.
8) Performance Observation: The implementation records execution time for communication monitoring, request classification, and privacy-preserving processing stages. The collected measurements are displayed to support observation of the overall communication workflow and processing behavior.

b) Methods/Technologies:

1. HMAC-Based Device Authentication: Hash-based Message Authentication Code (HMAC) was employed as the identity generation and authentication mechanism for IoT devices. HMAC generates secure identity values using device-specific information and secret parameters to verify communication legitimacy. In this project, unique secret identities were generated for IoT devices and stored for validation. During communication initialization and session establishment, generated identities were matched against stored values to authenticate devices before allowing communication.

2. Chaotic Privacy Mechanism (Chaotic S-Box Transformation): A chaotic privacy mechanism was implemented to support lightweight privacy preservation during communication. The technique utilizes hash-derived values and iterative transformation processes to generate message-dependent substitution values used during communication handling. In this project, chaotic processing was applied as an additional privacy-oriented layer before communication execution to strengthen message protection while maintaining lightweight operation suitable for heterogeneous IoT environments.

3. Neural Network-Based Attack Detection: A neural-network-based classification model was incorporated to monitor communication behavior and identify suspicious requests. Communication events and device-related attributes were converted into structured inputs and processed for classification. The implemented model consists of multiple dense processing layers and performs prediction to categorize requests as normal or attack-related communication. This component supports communication validation during authentication and session establishment stages.

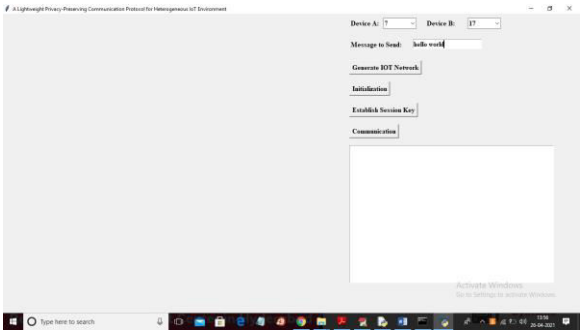
4. Symmetric Encryption (Fernet Encryption): Symmetric encryption was used to secure message transmission between communicating IoT devices. The implemented encryption mechanism performs message encryption before transmission and decrypts received data at the destination device using a shared secret key. In this project, encrypted communication occurs after successful authentication and session establishment, enabling protected end-to-end message exchange through the Control Center.

5. Communication Logging and Data Processing: Communication logging and structured data processing were used to record communication events and prepare inputs for monitoring activities. Event records were generated during authentication and communication stages and converted into structured

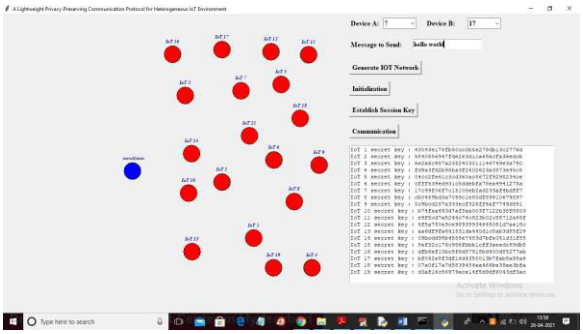
representations suitable for analysis. Basic preprocessing operations including label encoding and transformation were applied to communication data before prediction, enabling automated communication assessment and secure interaction management.

4. EXPERIMENTAL RESULTS

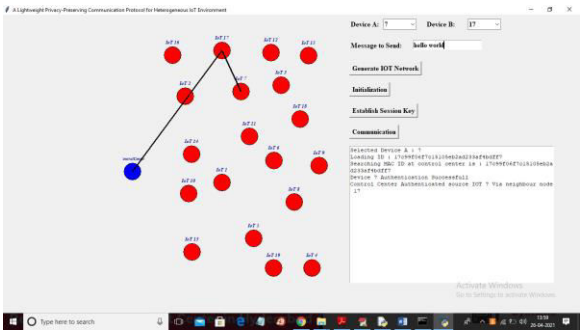
To run project double click on 'run.bat' file to get below screen



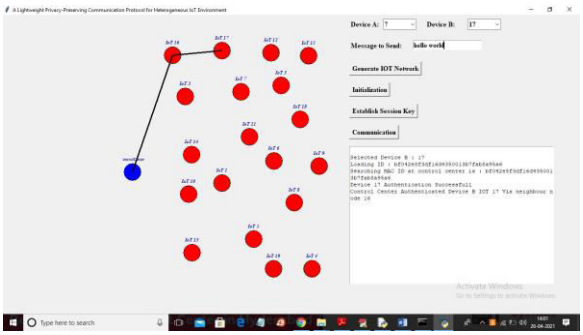
In above screen from drop down box select device A and B and then enter some message and I selected device A as 7 and device B as 17 and entered message as 'hello world' and now click on 'Generate IOT Network' button to generate IoT network



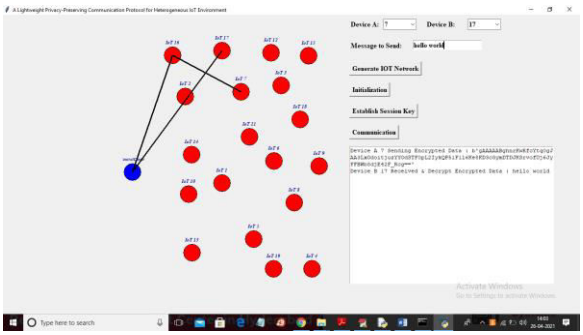
In above screen IoT network generated and all red colour circles are normal IoT and blue colour circle is the Control Center and in text area we can see secret identity of key generated for each IoT. Now click on 'Initialization' button to perform authentication between device A and B



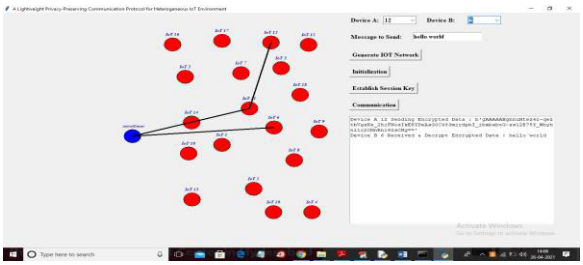
In above screen IoT 7 is getting authenticated from Control Center by sending message to it and once after successful authentication then we can see device A dataset base ID and generated ID both matched. Now click on 'Establish Session Key' button to establish session communication key between device A, Control Center and device B



In above screen Control Center authenticating Device B also and once after authentication then session key will be establish. Now click on 'Communication' button to send data between Device A, Control Center and Device B



In above screen Device A sending data to neighbour IoT 16 and then IoT sending data to Control Center and then Control Center sending data to IoT17. In above screen text area we can see device A sent encrypted data and then device B received encrypted and perform decryption to get original data. Similarly you can select any source or destination device and then Control Center perform authentication and transfer data between them



In above screen we can see IoT12 is sending data to IoT 6 by using neighbour IoT and Control Center.

5. CONCLUSION

In this work, we introduced an ultra-lightweight device-to-device security protocol based exclusively on a symmetric key-based scheme. Our protocol offers protection for heterogeneous Internet of Things environments. In this protocol, the synchronous key delegation mechanism is designed utilizing a chaotic system, namely Logistic Map, which guarantees the unpredictable, unrepeatable, and determinate properties of the symmetric keys. We evaluate the security and effectiveness of the proposed protocol, as well as its resistance to several detrimental vulnerabilities. The outcome demonstrates that our protocol for smart home systems outperforms previous symmetric key-based work.

Future enhancements can focus on integrating the lightweight communication protocol with emerging IoT paradigms such as edge computing, 5G/6G networks, and AI-driven security frameworks. The protocol can be extended to support large-scale heterogeneous environments with dynamic device mobility and real-time threat detection. Further investigation may include resistance against advanced cyberattacks, blockchain-based trust management, and secure multi-device authentication. Additionally, optimizing key management for ultra-low-power sensors and validating performance in real-world smart city and connected vehicle deployments can improve scalability, reliability, and practical adoption.

REFERENCES

[1] T. Song, R. Li, B. Mei, J. Yu, X. Xing, and X. Cheng, ‘‘A privacy preserving communication protocol for IoT applications in smart homes,’’ IEEE Internet Things J., vol. 4, no. 6, pp. 1844–1852, Dec. 2017, doi: 10.1109/JIOT.2017.2707489.

[2] J. Qiu, L. Du, D. Zhang, S. Su, and Z. Tian, ‘‘Nei-TTE: Intelligent traffic time estimation based on fine-grained time derivation of road segments for smart city,’’ IEEE Trans Ind. Informat., vol. 16, no. 4, pp. 2659–2666, Apr. 2020.

[3] Z. Tian, X. Gao, S. Su, and J. Qiu, ‘‘Vcash: A novel reputation framework for identifying denial of traffic service in Internet of connected vehicles,’’ IEEE Internet Things J., to be published.

[4] Sumsung Smartthings Developers Documentation. [Online]. Available: https://smartthings.developer.samsung.com/blog/en-us/2019/ 01/17/Shape-the-Future-of-IoT-with-SmartThings

[5] J. Van Bulck, J. T. Mühlberg, and F. Piessens, ‘‘VulCAN: Efficient component authentication and software isolation for automotive control networks,’’

in Proc. 33rd Annu. Comput. Secur. Appl. Conf. (ACSAC), Orlando, FL, USA, Dec. 2017, pp. 225–237, doi: 10.1145/3134600.3134623.

[6] R. L. Rivest, A. Shamir, and L. Adleman, ‘‘A method for obtaining digital signatures and public-key cryptosystems,’’ Commun. ACM, vol. 21, no. 2, pp. 120–126, Feb. 1978, doi: 10.1145/359340.359342.

[7] W. Diffie and M. Hellman, ‘‘New directions in cryptography,’’ IEEE Trans. Inf. Theory, vol. IT-22, no. 6, pp. 644–654, Nov. 1976, doi: 10.1109/TIT.1976.1055638.

[8] C. A. Lara-Nino, A. Diaz-Perez, and M. Morales-Sandoval, ‘‘Elliptic curve lightweight cryptography: A survey,’’ IEEE Access, vol. 6, pp. 72514–72550, 2018, doi: 10.1109/ACCESS.2018.2881444.

[9] S. Kumar, Y. Hu, M. P. Andersen, R. A. Popa, and D. E. Culler, ‘‘JEDI: Many-to-many end-to-end encryption and key delegation for IoT,’’ in Proc. 28th USENIX Secur. Symp., USENIX Secur., Santa Clara, CA, USA, Aug. 2019, pp. 1519–1536. [Online]. Available: https://www.usenix.org/conference/usenixsecurity19/ presentation/kumarsam

[10] A. K. Das, S. Zeadally, and D. He, ‘‘Taxonomy and analysis of security protocols for Internet of Things,’’ Future Gener. Comput. Syst., vol. 89, pp. 110–125, Dec. 2018, doi: 10.1016/j.future.2018.06.027.

[11] L. Eschenauer and V. D. Gligor, ‘‘A key-management scheme for distributed sensor networks,’’ in Proc. 9th ACM Conf. Comput. Commun. Secur. (CCS), Washington, DC, USA, Nov. 2002, pp. 41–47, doi: 10.1145/586110.586117.

[12] H. Chan, A. Perrig, and D. Song, ‘‘Random key predistribution schemes for sensor networks,’’ in Proc. Symp. Secur. Privacy, Berkeley, CA, USA, May 2003, pp. 197–213, doi: 10.1109/SECPRI.2003. 1199337.

[13] C. Blundo, A. De Santis, A. Herzberg, S. Kutten, U. Vaccaro, and M. Yung, ‘‘Perfectly-secure key distribution for dynamic conferences,’’ in Proc. 12th Annu. Int. Cryptol. Conf., Adv. Cryptol. (CRYPTO), Santa Barbara, CA, USA, Aug. 1992, pp. 471–486, doi: 10.1007/ 3-540-48071-4_33.

[14] D. Liu, P. Ning, and R. Li, ‘‘Establishing pairwise keys in distributed sensor networks,’’ ACM Trans. Inf. Syst. Secur., vol. 8, no. 1, pp. 41–77, Feb. 2005, doi: 10.1145/1053283.1053287.

[15] A. K. Das, ‘‘A random key establishment scheme for multi-phase deployment in large-scale distributed sensor networks,’’ Int. J. Inf. Secur., vol. 11, no. 3, pp. 189–211, Apr. 2012, doi: 10.1007/s10207-012-0162-9.

[16] F. Hendaoui, H. Eltaief, and H. Youssef, ‘‘A collaborative key management scheme for distributed smart objects,’’ Trans. Emerg. Telecommun. Technol., vol. 29, no. 6, Jun. 2018, Art. no. e3198, doi: 10.1002/ ett.3198.

[17] A. K. Das, ‘‘ECPKS: An improved location-aware key management scheme in static sensor networks,’’ Int. J. Netw. Secur., vol. 7, no. 3, pp. 358–369, 2008. [Online]. Available: http://ijns.femto.com.tw/ contents/ijns-v7-n3/ijns-2008-v7-n3-p358-369.pdf

[18] I.-C. Tsai, C.-M. Yu, H. Yokota, and S.-Y. Kuo, ‘‘Key management in Internet of Things via kronecker product,’’ in Proc. IEEE 22nd Pacific Rim Int. Symp. Dependable Comput. (PRDC), Christchurch, South

Island, Jan. 2017, pp. 118–124, doi: 10.1109/PRDC.2017.25.

[19] T. Song, R. Li, B. Mei, J. Yu, X. Xing, and X. Cheng, ‘‘A privacy preserving communication protocol for IoT applications in smart homes,’’ in Proc. Int. Conf. Identificat., Inf. Knowl. Internet Things (IIKI), Beijing, China, Oct. 2016, pp. 519–524, doi: 10.1109/IIKI.2016.3.

[20] W. S. Sayed, A. G. Radwan, and H. A. H. Fahmy, ‘‘Design of a generalized bidirectional tent map suitable for encryption applications,’’ in Proc. 11th Int. Comput. Eng. Conf. (ICENCO), Dec. 2015, pp. 207–211.

[21] T. S. Chaware and B. K. Mishra, ‘‘Secure communication using TPC and chaotic encryption,’’ in Proc. Int. Conf. Inf. Process. (ICIP), Dec. 2015, pp. 615–620.

[22] P. Tobin, L. Tobin, M. McKeever, and J. Blackledge, ‘‘Chaos-based cryptography for cloud computing,’’ in Proc. 27th Irish Signals Syst. Conf. (ISSC), Jun. 2016, pp. 1–6.

[23] C. Hu, A. Alhothaily, A. Alrawais, X. Cheng, C. Sturtivant, and H. Liu, ‘‘A secure and verifiable outsourcing scheme for matrix inverse computation,’’ in Proc. IEEE IEEE Conf. Comput. Commun. (INFOCOM), Atlanta, GA, USA, May 2017, pp. 1–9, doi: 10.1109/INFOCOM.2017. 8057199.

[24] S. A. Hirani, ‘‘Energy consumption of encryption schemes in wireless devices,’’ Ph.D. dissertation, Univ. Pittsburgh, Pittsburgh, PA, USA, 2003.

[25] W. Liao, C. Luo, S. Salinas, and P. Li, ‘‘Efficient secure outsourcing of large-scale convex separable programming for big data,’’ IEEE Trans. Big Data, vol. 5, no. 3, pp. 368–378, Sep. 2019, doi: 10.1109/TBDATA.2017.2787198.

[26] D. Dolev and A. Yao, ‘‘On the security of public key protocols,’’ IEEE Trans. Inf. Theory, vol. 29, no. 2, pp. 198–208, Mar. 1983, doi: 10.1109/TIT.1983.1056650.

[27] C. J. F. Cremers, S. Mauw, and E. P. de Vink, ‘‘Injective synchronisation: An extension of the authentication hierarchy,’’ Theor. Comput. Sci., vol. 367, nos. 1–2, pp. 139–161, Nov. 2006, doi: 10.1016/j.tcs.2006.08.034.

[28] S. Meier, C. Cremers, and D. Basin, ‘‘Strong invariants for the efficient construction of machine-checked protocol security proofs,’’ in Proc. 23rd IEEE Comput. Secur. Found. Symp., Edinburgh, U.K., Jul. 2010, pp. 231–245, doi: 10.1109/CSF.2010.23.

[29] J. Daemen and V. Rijmen, The Design of Rijndael: AES—The Advanced Encryption Standard (Information Security and Cryptography). Springer, 2002, doi: 10.1007/978-3-662-04722-4.

[30] D. Eastlake, III, and P. E. Jones, US Secure Hash Algorithm 1 (SHA1), document RFC 3174, 2001, pp. 1–22, doi: 10.17487/RFC3174.