

Enhanced n-party Diffie Hellman key exchange algorithm using the divide and conquer algorithm

Nwanze Chukwudi Ashioba, Patrick Ogholorunwalomi Ejeh, Azaka Maduabuchuku

Department of Computer Science, Faculty of Computing, Dennis Osadebay University, Asaba, Nigeria

Article Info

Article history:

Received Jul 25, 2024

Revised Dec 4, 2024

Accepted Dec 15, 2024

Keywords:

Asymmetric cryptography

Cryptography

Cryptosystem

Key exchange

Private key

Symmetric cryptography

ABSTRACT

Cryptographic algorithms guarantee data and information security via a communication system against unauthorized users or intruders. Numerous encryption techniques have been employed to safeguard this data and information from hackers. By supplying a distinct shared secret key, the n-party Diffie Hellman key exchange approach has been used to protect data from hackers. Using a quadratic time complexity, the n-party Diffie-Hellman method is slow when multiple users use the cryptographic key interchange system. To solve this issue, the researchers created an effective shared hidden key for the n-party Diffie Hellman key exchange of a cryptographic system using the divide-and-conquer strategy. The current research recommends the use of the divide and conquer algorithm, which breaks down the main problem into smaller subproblems until it reaches the base solution, which is then merged to generate the solution of the main problem. The comparative analysis indicates that the developed system generates a shared secret key faster than the current n-party Diffie Hellman system.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Nwanze Chukwudi Ashioba

Department of Computer Science, Faculty of Computing, Dennis Osadebay University

Asaba, Delta State, Nigeria

Email: nwanze.ashioba@dou.edu.ng

1. INTRODUCTION

When conveyed from one region to a different location or region within the context of communication, data, and information are always vulnerable to danger and attackers. Several cryptographic techniques have been used to secure these data and information. Ashioba and Yoro [1] presented cryptography as a technique for confirming the confidentiality and authenticity of data. The Greek terms “kryptos,” which means undisclosed, and “graphy,” which means writing, are the root of cryptography [2]. It's a scientific technique for encrypting messages to keep hackers away. Information can be modified scientifically using cryptography to thwart attacks [3]. Essentially, cryptography is the process of encrypting data to guard against intrusion over shaky communication channels [4], [5]. It mostly makes use of the mathematical idea that produces a variety of procedures known as cryptographic algorithms [6]. The gathering of cryptographic methods as well as the key management procedures that enable the use of these techniques in every kind of environment are referred to as cryptography in this sense [7].

Asymmetric key cryptography and symmetric key cryptosystems are the two types of cryptography [8], [9]. Cryptosystems that use asymmetric key cryptography use separate keys for decryption and encryption [10]. There is a mathematical relationship between the keys [11]. An asymmetric key cryptosystem, often known as a public key cryptosystem, uses both private and public keys to encrypt and decrypt data. The fact that each user's visible key is made hidden prevents users from sharing or disclosing private information (keys). A common secret key is used in symmetric key cryptography to encrypt and

decrypt data and information [10]. For symmetric key cryptography to operate correctly and efficiently, both the person sending the message and the receiver of the message must be mindful of and use the same confidential key when communicating [12]. Reaching an agreement on a lone hidden key that only the parties concerned are aware of is the trial with symmetric key cryptography. In 1977, Whitefield Diffie and Martin Hellman published a paper suggesting a key interchange mechanism as a remedy to this problem, proposing the first feasible result. In the Diffie-Hellman key exchange technology participants communicate over an open channel to determine a common key without disclosing the confidential keying information earlier [12].

The Diffie-Hellman key exchange, also known as exponential key exchange, is a technique for securely swapping cryptographic keys over an erratic channel. It is the basic construction block of numerous protected communication protocols including secure sockets layer (SSL), transport layer security (TLS), and secure shell (SSH). An innovation in public key cryptography was the Diffie-Hellman key interchange mechanism that permits participants to strongly produce a collective hidden key for interaction.

To ensure multicast fidelity, the procedure created a common underground key for a cluster of participants. Various plans have been put out over time. The researches [13], [14] created an effective Diffie-Hellman-MAC key exchange system by employing a message authentication code (MAC) hash function. Restructured internet architecture by [15] reduced the possibility of stealthy network attacks. By adding more security codes to the current method, they improved the encryption protocol's security. Jha and Patil [16] created a Diffie-Hellman algorithm enhancement. To ensure the security of the transmission, they implemented specific mathematical techniques. Adrian *et al.* [17] created a blowfish encryption technique-based version of the Diffie-Hellman key exchange algorithm for network security.

2. LITERATURE REVIEW

Pathak and Sanghi [18] created a two-password-based simple three-party key exchange protocol via the twin-Diffie-Hellman algorithm. The algorithm provided greater security and efficiency than the computational-based Diffie-Hellman protocol. The protocols were verified using automated validation of internet security protocol and application (AVISPA).

A password-based key interchange mechanism was anticipated by [19] and it was used to secure communication between participants. Each party in this protocol computes a shared secret key with the other parties using a password. A parallel Diffie-Hellman key exchange (PDHKE) protocol was suggested by Tseung and Wu in 2008. For three or more participants, the protocol was an expansion of the regular Diffie-Hellman key exchange. Using the Diffie-Hellman key exchange, each party in the protocol computes a shared secret key with every other party.

A group Diffie-Hellman key exchange protocol for two or more participants was created by [20]. For two or more participants, the protocol is an extension of the conventional Diffie-Hellman key exchange. Under this protocol, the Diffie-Hellman key exchange is employed by each party to compute a shared secret key with all other parties. The researches [21], [22] presented a methodology that used an interactive zero-knowledge proof to alter the Diffie-Hellman key exchange algorithm. The communication system's known attacks were thwarted by the protocol. Alam [23] developed an improved key exchange protocol based on a third-party authentication scheme which eliminated the man-in-middle attack on the Diffie-Hellman key exchange protocol. Wu [24] created the verifier-based n-party password-authenticated key exchange (PAKE) protocol, which secures the exchange of password authentication keys. The n-party communicated over an unprotected channel using the protocol. In this protocol, the shared secret key is computed by each party with every other party using a password, and it is validated by a verifier.

A technique that facilitates secure communication between several parties is the PDHKE mechanism, wherein every party uses the same exponent to produce peer-to-peer keys [25]. In addition, the Diffie-Hellman key exchange has been instrumental in the development of group key establishment methods that are customized for a variety of situations, including multicast groups and ad hoc networks. Based on the two-party Diffie-Hellman technique, methods such as the group-Diffie-Hellman protocol have been proposed to improve contributory group key exchange [26]. Rimani *et al.* [27] developed an image registration with key fourier transform for Diffie-Hellman key exchange protocol. The algorithm created a transformation between images for recovering the key by the receiver.

Furthermore, dynamic group Diffie-Hellman protocols are designed to accommodate circumstances in which group membership varies, allowing participants to enter and quit the group with ease at any time. Protecting security and efficiency is still essential for modern Diffie-Hellman key exchange systems. In resource-constrained contexts such as location-aided mobile ad-hoc networks, techniques such as employing elliptic curve Diffie-Hellman instead of conventional Diffie-Hellman have been investigated to improve efficiency [28]. Additionally, techniques such as utilizing the Diffie-Hellman method in third party auditor (TPA) interactions have been proposed to improve auditing performance, and the Diffie-Hellman key exchange has been found incorporates in cloud storage auditing [29]. Francis *et al.* [30] proposed the Diffie-

Hellman key exchange algorithm based on image data encryption to protect the privacy and confidentiality of sensitive data over unreliable channels. The researches [31], [32] improved confidentiality, integrity, authentication, and privacy in the context of the internet of things (IoT) by integrating the elliptic curves (ECG) and the traditional public key infrastructure.

3. RESEARCH METHOD

3.1. N-party Diffie-Hellman conceptual framework

The conceptual framework of the n-party Diffie-Hellman algorithm with three participants is shown in Figure 1. Each of the three participants has their own set of private and public keys. The private keys are kept secret, but the public key is exchanged among the users in the communication.

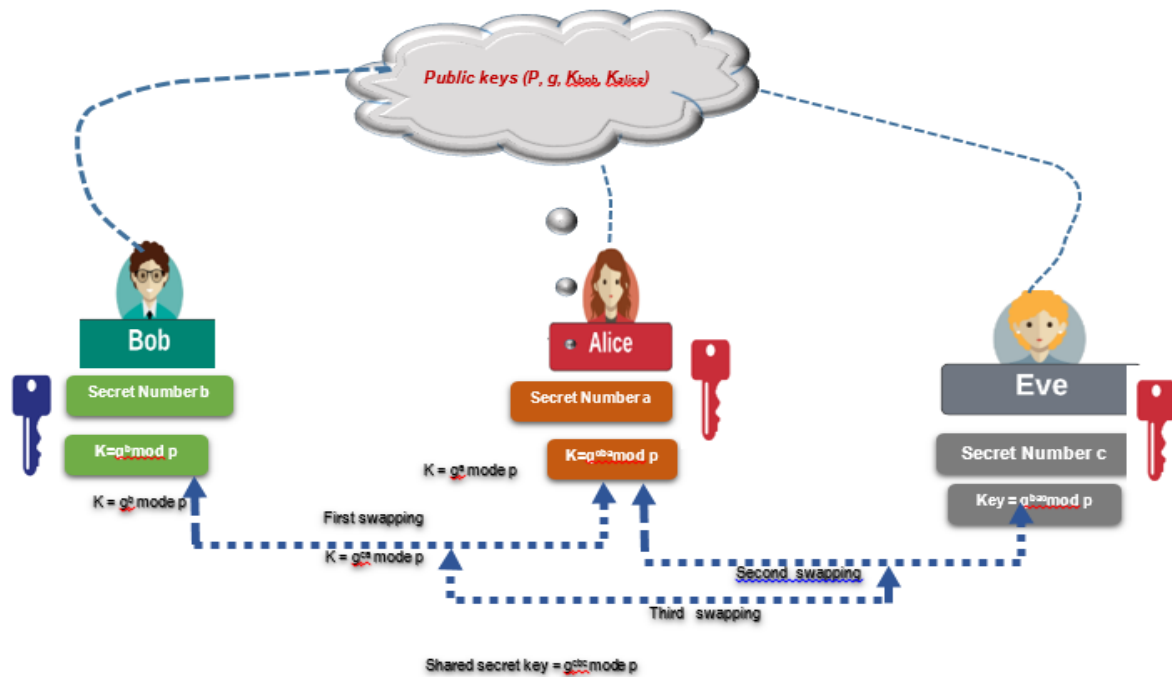


Figure 1. Framework of the n-party Diffie-Hellman algorithm

The diagram in Figure 1 shows that the number of swapping operations for three participants is 3. In (1) generally indicates the number of n-party Diffie-Hellman algorithm swapping operations for n participants [3].

$$\text{No of Swapping operation} = \frac{n(n-1)}{2} = \frac{n^2+n}{2} \quad (1)$$

Therefore, the time complexity for the n-parties Diffie-Hellman algorithm is the quadratic relationship shown by (2).

$$T_n = O(n^2) \quad (2)$$

The algorithm takes plenty of time to compute the number of swapping operations in the process when the number of participants is large.

3.2. Algorithm of the n-party Hellman key exchange approach

In a cryptographic system, an algorithm lays out instructions that describe how to generate the shared secret key consecutively [33]. The steps include:

- i) The parties agreed on two positive numbers, p and q.
- ii) A private key is selected at random by each party, say x_i .

iii) Each party calculates the public key.

$$K_a = q^x \bmod p, K_b = q^y \bmod p, \dots, K_n = q^n \bmod p$$

iv) To compute the common shared secret key, all parties swap or exchange their public keys.

$$K_{ab} = q^{xy \dots n} \bmod p$$

3.3. Divide and conquer algorithm

A problem-solving strategy known as the divide and conquer algorithm, approach, or principle works by breaking the main problem down into smaller problems that are subsequently further divided into smaller problems that are solved separately, and then merging or combining them to find the solution to the original problem [34]. The recursive equation of the divide and conquer approach is shown in (3).

$$T_n = T_{\frac{n}{2}} + O(C) \quad (3)$$

Where n is the number of participants in the communication system, $T_{\frac{n}{2}}$ is the time taken to compute the solutions of the sub problems, and T_n is the time taken to compute the solution of the main problem. The time complexity of the algorithm is equal to the number of divide-and-conquer operations performed by the n -party and is shown in (4).

$$T_n = O(\log_2 n) \quad (4)$$

3.4. Conceptual design of n-party Diffie-Hellman algorithm using divide and conquer algorithm

The conceptual framework of the n-party Diffie-Hellman using the divide and conquer algorithm to generate the shared secret key in a communication system is illustrated in Figure 2. It contains three participants in the communication system, where each party has a private key that is kept secret and a public key that is generated using modular exponentiation. The participants' public keys were divided into sub-participants until the base participant was reached. The solutions of the base participants are merged to obtain the solution to the main problem (shared secret key). Figure 2 shows that the number of swapping operations performed by the participants in generating the shared secret key, using the divide and conquer approach, is 2. Therefore, the time complexity of the divide and conquer algorithm in generating the shared secret key of the Diffie-Hellman algorithm of n participants is equal to the number of divide and conquer operations performed by the n -party and is shown in (4).

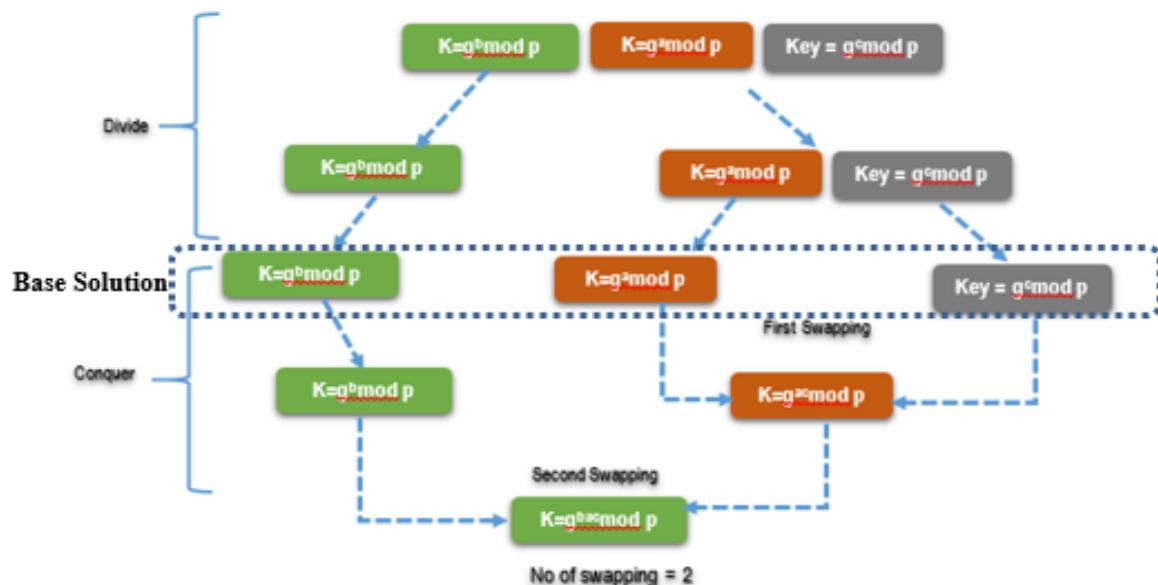


Figure 2. Framework of the divide and conquer algorithm in n-party Diffie-Hellman approach

3.5. Algorithm of the proposed system

The steps of the proposed system include:

- i) The parties agreed on two positive numbers, p and q .
- ii) A private key is selected at random by each party, say x_i .
- iii) Each party calculates the public key.

$$K_a = q^x \bmod p, K_b = q^y \bmod p, \dots, K_n = q^n \bmod p$$

- iv) The parties are divided into two (2) sub-parties using recursion.
- v) Solve the smaller sub-parties recursively to find the base solution.
- vi) Combine the solutions of the sub-parties recursively to find the shared secret key $K_{ab} = q^{xy \dots n} \bmod p$.

3. RESULTS AND DISCUSSION

Table 1 presents the data collected, and the results obtained from the analysis of the algorithms for 20 participants in the communication system. Table 1 is presented graphically in Figure 3. The result from Figure 3 shows that the divide and conquer approach takes less time to generate a shared secret key than the Diffie-Hellman algorithm when the number of participating parties is very large. Since the performance of a system is inversely proportional to the time taken, the relationship is shown mathematically in (5).

Table 1. Time complexity analysis between quadratic and logarithmic Diffie-Hellman algorithms

Input size n	N-partyDiffie-Hellman algorithm	Diffie-Hellman algorithm using divide and conquer algorithm
2	4,000	1,000
3	9,000	1,585
4	16,000	2,000
5	25,000	2,322
6	36,000	2,585
7	49,000	2,807
8	64,000	3,000
9	81,000	3,170
10	100,000	3,322
11	121,000	3,459
12	144,000	3,585
13	169,000	3,700
14	196,000	3,807
15	225,000	3,907
16	256,000	4,000
17	289,000	4,087
18	324,000	4,170
19	361,000	4,248
20	400,000	4,322

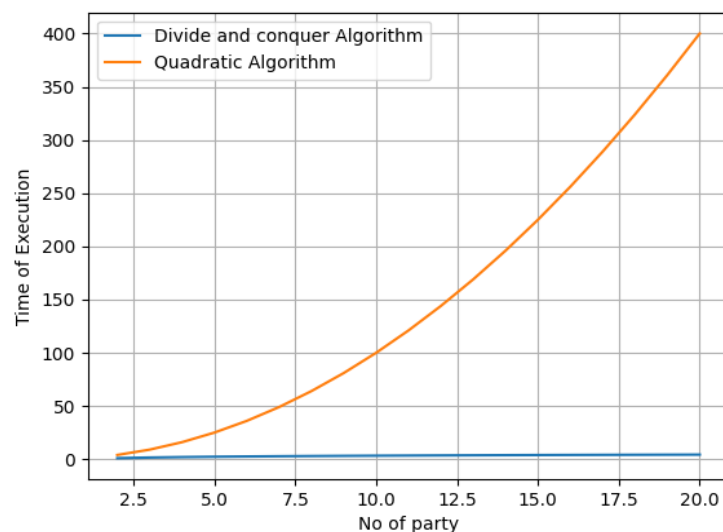


Figure 3. Analysis of the execution time of the Diffie-Hellman algorithm

$$\frac{P_A}{P_B} = \frac{T_B}{T_A} \quad (5)$$

Where:

P_A = performance of the logarithmic Diffie-Hellman key exchange approach.

P_B = performance of the quadratic Diffie-Hellman key exchange algorithm.

T_A = time taken for n participant in the logarithmic Diffie-Hellman key exchange approach.

T_B = time taken for n participants in the quadratic Diffie-Hellman key exchange algorithm.

With 20 participants, we have:

$$\frac{P_A}{P_B} = \frac{T_B}{T_A} = \frac{400}{4.322} = 92.55 \approx 93$$

This shows that the logarithmic Diffie-Hellman key exchange approach performs 93 times better than the quadratic Diffie-Hellman key exchange approach in generating the shared secret key in a communication system.

4. CONCLUSION

Recent observations show that the n-party Diffie-Hellman key exchange approach has been used to generate the shared secret key in a communication system. From the study it was observed that the n-party Diffie-Hellman key exchange approach takes plenty of time to generate the shared secret in a communication system. This study compares the performance of the logarithmic key exchange technique with the time parameter against the quadratic key exchange algorithm. Our findings prove that the logarithmic Diffie-Hellman key exchange approach is faster than the quadratic Diffie-Hellman key exchange approach in generating the shared secret key of large participants in a communication system.

ACKNOWLEDGEMENTS

The writers would like to express their gratitude to friends and family for their financial and spiritual support in getting their article published.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Nwanze Chukwudi	✓	✓	✓	✓	✓	✓		✓	✓	✓			✓	✓
Ashioba														
Patrick		✓				✓		✓	✓	✓	✓	✓		
Ogholorunwalomi Ejeh														
Azaka Maduabuchuku	✓		✓	✓			✓			✓	✓		✓	

C : Conceptualization

M : Methodology

So : Software

Va : Validation

Fo : Formal analysis

I : Investigation

R : Resources

D : Data Curation

O : Writing - Original Draft

E : Writing - Review & Editing

Vi : Visualization

Su : Supervision

P : Project administration

Fu : Funding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

The authors confirm that the data supporting the findings of this study are available within the article [and/or its supplementary materials].




REFERENCES

- [1] N. C. Ashioba and R. E. Yoro, "RSA cryptosystem using object-oriented modeling technique," *International Journal of Information and Communication Technology Research*, vol. 4, no. 2, pp. 57–61, 2014.
- [2] D. Talukdar and P. L. P. Saikia, "A review on different encryption techniques : a comparative study," *International Journal of Engineering Research and General Science*, vol. 3, no. 3, pp. 1622–1625, 2015.
- [3] B. A. Forouzan, "Cryptography," in *Data Communication and Networking*, 4th Editio., New York, 2008, pp. 931–960.
- [4] A. Kaushik and Satvika, "Extended diffie-hellman algorithm for key exchange and management," *Proceedings of 2nd International Conference on Emerging Trends in Engineering and Management*, vol. 3, no. 3, pp. 67–70, 2013.
- [5] S. Boni, J. Bhatt, and S. Bhat, "Improving the Diffie-Hellman key exchange algorithm by proposing the multiplicative key exchange algorithm," *International Journal of Computer Applications*, vol. 130, no. 15, pp. 7–10, Nov. 2015, doi: 10.5120/ijca2015907170.
- [6] M. A. Chavan, M. A. Jadhav, M. S. Kumbhar, M. I. Joshi, and M. I. Joshi, "Data transmission using RSA algorithm," *International Research Journal of Engineering and Technology*, pp. 2008–2010, 2019, [Online]. Available: www.irjet.net.
- [7] O. Abari, J. Shola, and S. Philip, "Comparative analysis of discrete logarithm and rsa algorithm in data cryptography," *International Journal of Computer Science and Information Security*, vol. 13, pp. 24–31, 2015.
- [8] M. F. Mushtaq, S. Jamah, A. H. Disina, Z. D. Pinda, N. S. Shakir, and M. M. Deris, "Review on comparative study of various cryptography algorithm," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 5, no. 4, pp. 1–8, 2015.
- [9] M. Faheem, S. Jamel, A. Hassan, Z. A., N. Shafinaz, and M. Mat, "A survey on the cryptographic encryption algorithms," *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 11, pp. 333–344, 2017, doi: 10.14569/ijacsa.2017.081141.
- [10] N. Kaur and R. Nagpal, "Authenticated Diffie-Hellman key exchange algorithm," *International Journal of Computer Science and Information Technologies*, vol. 5, no. 4, pp. 5404–5407, 2014, [Online]. Available: www.ijcsit.com.
- [11] S. M. Seth and R. Mishra, "Comparative analysis of encryption algorithms for data communication," *Ijcsst*, vol. 2, no. 2, pp. 292–294, 2011.
- [12] R. College, "An approach to public-key cryptography using Diffie-Hellman key exchange algorithm," *International Journal for Research in Engineering Application & Management (IJREAM)*, no. 08, pp. 69–75, 2017.
- [13] E. J. Yoon and K. Y. Yoo, "An efficient Diffie-Hellman-MAC key exchange scheme," *2009 4th International Conference on Innovative Computing, Information and Control, ICICIC 2009*, pp. 398–400, 2009, doi: 10.1109/ICICIC.2009.80.
- [14] N. Li, "Research on diffie-hellman key exchange protocol," in *ICCET 2010 - 2010 International Conference on Computer Engineering and Technology, Proceedings*, 2010, vol. 4, pp. V4-634-V4-637, doi: 10.1109/ICCET.2010.5485276.
- [15] Vinothini, Saranya, and Vasumathi, "A study on Diffie-Hellman algorithm in network security," *International Journal Of Engineering And Computer Science*, vol. 3, pp. 7346–7349, 2014, [Online]. Available: www.ijecs.in.
- [16] M. Jha and S. Patil, "Advancement in Diffie-Hellman algorithm," *Journal of Engineering Research and Applications*, vol. 5, no. 7, pp. 01–02, 2015.
- [17] A. Adrian, M. Cendana, and S. D. H. Permana, "Diffie-Hellman key exchange modification using blowfish algorithm to prevent logjam attack," *Journal of Telecommunication, Electronic and Computer Engineering*, vol. 10, no. 4, pp. 1–7, 2018.
- [18] H. K. Pathak and M. Sanghi, "Simple three-party key exchange protocol via twin Diffie-Hellman problem," *International Journal of Network Security*, vol. 15, no. 4, pp. 256–264, 2013.
- [19] J. W. Byun and D. H. Lee, "N-party encrypted Diffie-Hellman key exchange using different passwords," in *Lecture Notes in Computer Science*, vol. 3531, 2005, pp. 75–90.
- [20] G. P. Biswas, "Diffie-Hellman technique: extended to multiple two-party keys and one multi-party key," *IET Information Security*, vol. 2, no. 1, pp. 12–18, Mar. 2008, doi: 10.1049/iet-ifs:20060142.
- [21] M. K. Ibrahim, "Modification of Diffie-Hellman key exchange algorithm for zero knowledge proof," *Engineering and Technology Journal*, vol. 30, no. 3, pp. 443–453, Jan. 2012, doi: 10.30684/etj.30.3.9.
- [22] C. B. Prakash and S. Shavali, "FPGA implementation Diffie-Hellman key exchange algorithm using DES," *International Journal of Innovative Research in Electronics and Communications*, vol. 1, no. 4, pp. 26–36, 2014.
- [23] B. Alam, "Diffie-Hellman key exchange protocol with entities authentication," *International Journal Of Engineering And Computer Science*, vol. 6, no. 4, Apr. 2017, doi: 10.18535/ijecs/v6i4.06.
- [24] S. Wu, "Security analysis and enhancements of verifier-based password-authenticated key exchange protocols in the three-party setting," *Journal of Information Science and Engineering*, vol. 27, no. 3, pp. 1059–1072, 2011, doi: 10.1688/JISE.2011.27.3.16.
- [25] M. Manulis, "Group key exchange enabling on-demand derivation of peer-to-peer keys," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 5536 LNCS, pp. 1–19, 2009, doi: 10.1007/978-3-642-01957-9_1.
- [26] Y. M. Tseng and T. Y. Wu, "Analysis and improvement on a contributory group key exchange protocol based on the Diffie-Hellman technique," *Informatica*, vol. 21, no. 2, pp. 247–258, Jan. 2010, doi: 10.15388/informatica.2010.286.
- [27] R. Rimani, N. H. Said, A. Ali-Pacha, and O. Ozer, "Key exchange based on Diffie-Hellman protocol and image registration," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 21, no. 3, pp. 1751–1758, Mar. 2021, doi: 10.11591/ijeecs.v21.i3.pp1751-1758.
- [28] E. Bresson, O. Chevassut, and D. Pointcheval, "Provably authenticated group diffie-hellman key exchange – the dynamic case," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 2248, 2001, pp. 290–309.
- [29] R. K. Yarava and R. P. Singh, "Efficient and secure cloud storage auditing based on the Diffie-Hellman key exchange," *International Journal of Intelligent Engineering and Systems*, vol. 12, no. 3, pp. 50–58, Jun. 2019, doi: 10.22266/IJIES2019.0630.06.
- [30] M. S. Francis, J. D. Sweetlin, and M. Sandhiya, "Secure image communication: integrating Diffie-Hellman key exchange for enhanced confidentiality," in *2024 3rd International Conference on Artificial Intelligence for Internet of Things, AIoT 2024*, May 2024, pp. 1–6, doi: 10.1109/AIoT58432.2024.10574682.
- [31] A. Sebbah and K. Benamar, "A privacy-enhanced scheme within the public key infrastructure for the internet of things, employing elliptic curve Diffie-Hellman (ECDH)," *Indonesian Journal of Electrical Engineering and Informatics*, vol. 12, no. 1, pp. 65–74, Feb. 2024, doi: 10.52549/ijeei.v12i1.5392.
- [32] S. Mandal, S. Mohanty, and B. Majhi, "An ID-based authenticated three-party key exchange protocol," *ACCENTS Transactions on Information Security*, vol. 2, no. 7, pp. 62–72, Jan. 2017, doi: 10.19101/tis.2017.27002.




- [33] A. N. Chukwudi, E.-O. Obaro, O. C. U. O. Kpasa, and N. N. Daniel, "Modeling Diffie Hellman key exchange algorithm using object-oriented analysis and design technique," *International Journal on Cryptography and Information Security*, vol. 14, no. 2, pp. 01–08, Jun. 2024, doi: 10.5121/ijcis.2024.14201.
- [34] M. Z. Karim and N. Akter, "Optimum partition parameter of divide-and-conquer algorithm for solving closest-pair problem," *International Journal of Computer Science and Information Technology*, vol. 3, no. 5, pp. 211–219, Oct. 2011, doi: 10.5121/ijcsit.2011.3519.

BIOGRAPHIES OF AUTHORS






Nwanze Chukwudi Ashioba    earned his B.Sc. in computer science in 1995, his M.Sc. in computer science in 2007, and his Ph.D. in computer science in 2014, all from the University of Port-Harcourt, Rivers State. He is a lecturer in the Department of Computer Science at the Faculty of Computing at Dennis Osadebay University in Anwai-Asaba, Delta State. His research interests and specialization areas include software engineering, machine learning, and data science. He is a member of the Computer Professionals of Nigeria (CPN) and the Nigeria Computer Society. He can be contacted at email: nwanze.ashioba@dou.edu.ng.



Patrick Ogholorunwalomi Ejeh    received his HND in computer science from the Federal Polytechnic Auchi, Edo State in 2006; M.Sc. in computer science from Northumbria University, Newcastle, United Kingdom in 2010; and, his Ph.D. in computer science from Sunderland University, Sunderland, United Kingdom in 2017. He is currently a lecturer with the Department of Computer Science at Dennis Osadebay University, Asaba, Delta State. His research interests include; artificial intelligence, knowledge management, data science, and the internet of things. He is also a member Nigerian Computer Society and Higher Education Academic; United Kingdom. He can be contacted at this email: patrick.ejeh@dou.edu.ng.



Mr Azaka Maduabuchuku    received his M.Sc. and PGD from the National Open University of Nigeria after obtaining his HND in computer science from Nekede, Owerri. He is a lecturer in the Department of Computer Science, Dennis Osadebay University Asaba Delta State. His areas of specialization include computer networking, hardware maintenance, and programming. He can be contacted at email: azaka.maduabuchuku@dou.edu.ng.