International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Ensuring the Security and Privacy of Data in Wireless Sensor Intelligence Networks While Simultaneously Optimizing Usability and Efficacy

[1]Dr Parag Kalkar, [2]Gaurav Katoch, [3]Dr Rasna Sehrawat, [4]Dr. Deepali Rani Sahoo, [5]Dr. AR . Saravanakumar, [6]Arjun Singh, [7]Prof. Pankaj Kumar Mishra

Submitted: 28/10/2023 **Revised**: 15/12/2023 **Accepted**: 25/12/2023

Abstract: It is critical to ensure data privacy and security in wireless sensor intelligence networks (WSINs).A comprehensive strategy is needed to balance data security, privacy, usability, and efficacy. It is crucial to achieve the ideal balance between protecting confidential data and ensuring the WSIN serves its intended purpose while adhering to privacy and legal standards. However, can be balanced with optimizing usability and efficacy. This paper explores existing approaches to achieving this balance and proposes a HPSOGA Algorithm to get optimal solution to a variety of security issues by replicating natural behaviours and processes.

Keywords: *Wireless Network, Wireless Security, Wireless Threats, Wireless Privacy, Optimization*

I. Introduction

The concurrent optimization of usability and efficacy in wireless sensor intelligence networks poses a complex challenge in safeguarding data security and privacy. Wireless sensor networks (WSNs) are of the utmost importance in numerous fields, such as healthcare, smart cities, industrial automation, and environmental monitoring. The aim to achieve a harmonious coexistence of security, privacy, usability, and efficacy by examining a comprehensive synopsis of coding strategies:

- **Data Encryption:**

To encrypt data before transmission and decrypt it upon reception, use encryption libraries and algorithms. You can use standard cryptography libraries like OpenSSL or libraries unique to your sensor platform for this.

Example (Python with the Cryptography library):

"from cryptography.fernet import Fernet

Generate a key

key = Fernet.generate_key()

f = Fernet(key)

Encrypt and decrypt data

encrypted_data = f.encrypt(b"Sensitive data")

decrypted_data = f.decrypt(encrypted_data)"

- **Authentication and Authorization:**

Before authorizing sensor node access to the network, incorporate authentication mechanisms into the code to validate their identities. Each sensor's permitted operations should be governed by authorization logic.

Example (Python with Flask for web-based APIs):

"from flask import Flask, request

[1]Pro-Vice Chancellor, Savitribai Phule Pune University, Pune, Maharashtra.
[2]Assistant Professor, Jaypee Business School, JIIT Noida, Uttar Pradesh
[3]Assistant Professor, Amity Institute of Education, Amity University, Noida (U.P.), (https:// orcid.org/0000-0002-3184-2521).
[4]Assistant Professor, Symbiosis Law School, Noida, Symbiosis International (Deemed University),Pune
[5]Associate Professor, Department of Education, CDOE, Alagappa University, Karaikudi-630 003, Tamil Nadu
[6]Associate Professor, Department of Computer and Communication Engineering, Manipal University Jaipur, Rajasthan
[7]Professor, Glocal School of Science and Technology, Glocal University, Delhi-Yamunotri Marg, Saharanpur, U.P., India

```
from functools import wraps
app = Flask(__name)
def authenticate(func):
    @wraps(func)
    def wrapper(*args, **kwargs):
        # Implement authentication logic here
        if request.headers.get('Authorization') == 'Bearer Token123':
            return func(*args, **kwargs)
        else:
            return "Unauthorized", 401
    return wrapper
@app.route('/secure-endpoint')
@authenticate
def secure_resource():
    return "This is a secure resource"
if __name__ == '__main__':
    app.run()  "
```

- **Data Integrity:**

Implement checksums or hash functions for data integrity verification. For instance, prior to transmitting data, compute its SHA-256 hash and validate it at the receiving end.

Example (Python):

```
import hashlib
data = "Data to be sent"
checksum = hashlib.sha256(data.encode()).hexdigest()
# Send data and checksum together
```

- **Privacy Preservation:**

To ensure data anonymity, incorporate privacy-preserving algorithms such as differential privacy directly into your code.

Example (Python):

```
" from diffprivlib import mechanisms
# Create a differentially private Laplace mechanism
mech = mechanisms.Laplace()
# Add noise to sensitive data
noisy_data = mech.randomise(sensitive_data)  "
```

- **Secure Sensor Deployment:**

For the purpose of safeguarding your sensor nodes, implement tamper-evident hardware and code.

Example (Pseudo code):

```
"  if is_tampered():
    log_tampering_attempt()
    initiate_shutdown()  "
```

Protection of Sensitive Information:

WSINs often collect and transmit sensitive data, such as personal information, business data, or critical infrastructure information. To safeguard sensitive information from misuse, breaches, and unauthorized access, it is imperative to ensure data privacy and security "(Potlapally, N. R., Ravi, S., Raghunathan, A., & Jha, N. K. (2005))". Although wireless networking has many advantages, it also raises new security issues that change the information security risk profile of the company as a whole. Wireless security is primarily a management problem, even though the traditional reaction to threats and vulnerabilities in wireless security is the adoption of technological solutions. A comprehensive risk assessment in the context of the surrounding environment and the development of a plan to reduce known hazards are essential for the effective management of wireless technology hazards. We offer a framework to help managers comprehend and evaluate the many risks associated with using wireless technology. Additionally, we discuss a few possible strategies to mitigate these hazards "(Choi, M. K., Robles, R. J., Hong, C. H., & Kim, T. H. 2008)". To limit the hazards associated with wireless communication, wireless network security is an ongoing activity that requires a combination of technical solutions, user education, and policy enforcement. The particular precautions you must take will be determined by the nature and sensitivity of your wireless network, as well as the potential threats to it "(Sathyavani, K. S., & Selvi, P. (2014)". Below is the latest data from Google trends which show the insights related to brimming need for devising cryptography measures using strong

algorithms as treats are showing an increased trend over the time.

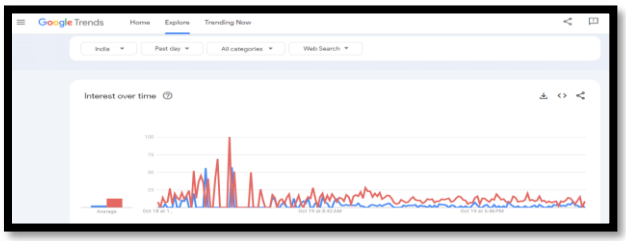

Figure 1 : Rising wireless security threats (Source: Google trends)

II. Background Studies

Wireless network security is essential for ensuring the confidentiality, integrity as well as availability of data carried via wireless networks. Wireless networks are vulnerable to a variety of threats and vulnerabilities, and countermeasures are required to limit these risks. Based on literature the researches discussed key elements related to WSINs "(Choi, M. K., Robles, R. J., Hong, C. H., & Kim, T. H. (2008))" .

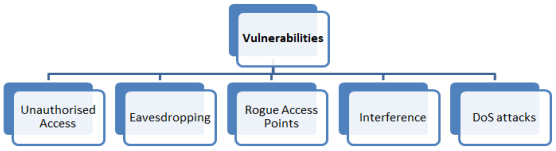

Figure 2 : Wireless network Vulnerabilities (source: "Choi, M. K., Robles, R. J., Hong, C. H., & Kim, T. H. (2008))"

Unauthorised Access: Because radio waves may penetrate walls and travel vast distances, wireless networks are vulnerable to unauthorised access. Unauthorised network connections can be attempted by attackers "(Choi, M. K., Robles, R. J., Hong, C. H., & Kim, T. H. (2008)" Eavesdropping: Wireless transmissions can be intercepted, allowing sensitive data to be eavesdropped on. Attackers can intercept and examine data packets as they go through the radio "(Choi, M. K., Robles, R. J., Hong, C. H., & Kim, T. H. (2008)". Rogue Access Points: Attackers can set up unauthorised or rogue access points, potentially causing users to connect to harmful networks inadvertently "(Choi, M. K., Robles, R. J., Hong, C. H., & Kim, T. H. (2008)". Interference: Because wireless networks are susceptible to interference from other devices or networks, communication can be disrupted "(Choi, M. K., Robles, R. J., Hong, C. H., & Kim, T. H. (2008)". DoS attacks: Attackers can flood a wireless network with traffic, rendering it inaccessible to normal users "(Choi, M. K., Robles, R. J., Hong, C. H., & Kim, T. H. (2008)".

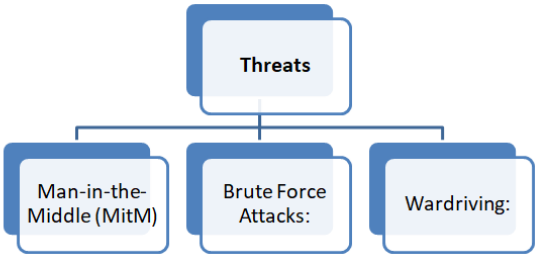

Figure 3 : Wireless network Threats(source: Noor, M. M., & Hassan, W. H. (2013).)

Man-in-the-Middle (MitM) attacks occur when an attacker positions himself between the client and the access point, intercepting and perhaps changing data (Noor, M. M., & Hassan, W. H. (2013)).**Brute Force Attacks:** Attackers can use exhaustive trial-and-error approaches to crack encryption keys. Attackers can transmit DE authentication or disassociation frames to disconnect legitimate users from the network (Noor, M. M., & Hassan, W. H. (2013)).**Wardriving:** Malicious individuals can drive about hunting for unprotected wireless networks to exploit. Attackers can inject malicious packets into the network, possibly jeopardising its integrity.

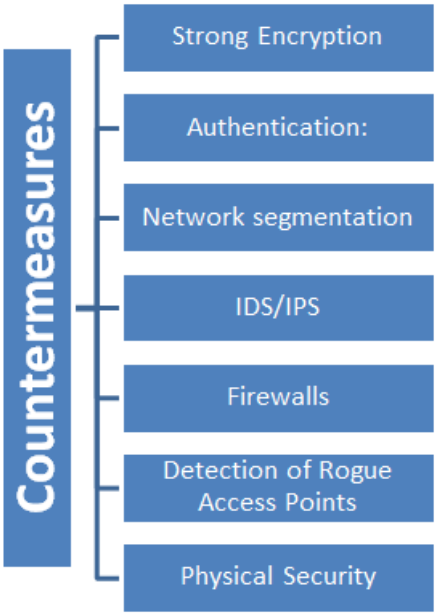

Figure 4 : Counter measures

Strong Encryption: To safeguard data while it's in transit, use strong encryption techniques like WPA3.

Steer clear of old and unsafe protocols such as WEP(Noor, M. M., & Hassan, W. H. (2013)).**Authentication:** To guarantee that only authorised users may access the network, utilise robust authentication methods including certificate-based authentication, biometrics, and complicated passwords(Noor, M. M., & Hassan, W. H. (2013)).**Network segmentation:** Divide the network into distinct zones and provide each section a different set of security controls. This may restrict the attack's potential range.Install intrusion detection and prevention systems (IDS/IPS) to identify and stop network anomalies and attacks(Noor, M. M., & Hassan, W. H. (2013)).**Firewalls:** To filter and manage incoming and outgoing traffic, use firewalls. Firewalls with statefulness can keep an eye on the status of open connections.**Detection of Rogue Access Points:** Continually search for and identify rogue access points nearby(Noor, M. M., & Hassan, W. H. (2013)).**Physical Security:** Equipment for networks that is physically secure(Noor, M. M., & Hassan, W. H. (2013)).

As per the literature there are many evolutionary algorithms. These algorithms, can seek optimal solutions to a variety of issues by replicating natural behaviours and processes. The algorithm has to be chosen is determined on the nature of the problem. Each of these techniques may be better suited to certain types of optimisation challenges. Following section discusses various optimization approaches found after doing extensive literature review.

Bee Colony Optimisation (BCO) which is inspired by honeybee foraging behaviour. It entails a swarm of artificial bees that look for solutions to optimisation challenges. Bees communicate via a network, just like ants leave pheromone trails **(Teodorović, D. (2009).** Particle Swarm Optimisation (PSO) which is Inspired by the social behaviour of birds and fish, PSO was developed. In PSO, a population of particles wanders around a solution space, altering their placements based on their own and their neighbours' experiences(Clerc and Kennedy, 2002). The Firefly Algorithm was inspired by the flashing patterns of fireflies. It simulates the attraction and repulsion of fireflies in order to solve an issue. Brighter firefly attract others, while dimmer ones gravitate towards brighter ones(**Yang, X. S., & He, X. (2013)).**The Bat Algorithm was inspired by the echolocation behaviour of bats. It mimics the movement of bats as they hunt for prey. Bats produce sound waves and alter their locations in response to the echo feedback **(Yang, X. S., & He, X. (2013))**. Although genetic algorithms are not directly inspired by social insects, they are a type of nature-inspired optimisation algorithm. To evolve superior answers across generations, they use a population of potential solutions and genetic operators such as mutation, crossover, and selection(**Holland, J. H. (1992))**. The GWO algorithm was inspired by grey wolf hunting behaviour. It optimises problem solutions by modelling the wolf pack's leadership hierarchy **(Mirjalili, S., Mirjalili, S. M., & Lewis, A. (2014))**.Cuckoo Search is based on the brood parasitism behaviour of several cuckoo species. It is used to solve optimisation problems and involves cuckoos discovering nests (solutions) in the nests of other bird species(**Yang, X. S., & Deb, S. (2014).).** The human immune system influenced AIS algorithms. They simulate the immune response in order to detect and destroy alien things, and they find use in optimisation and anomaly detection(**Dasgupta, D. (Ed.). (2012))**.

III. Proposed Algorithm

By iteratively updating particle locations and velocities based on their individual and global optimum positions, the PSO algorithm seeks to identify the optimal solution. It blends exploitation (by drawing attention to the best solutions thus far) and exploration (by randomly exploring the search space). PSO may be optimised for a variety of optimisation issues by changing the control parameters and initializations. (Clerc and Kennedy, 2002)

A popular method for solving optimisation and search issues is the Particle Swarm Optimisation (PSO) algorithm, which draws inspiration from nature (Clerc and Kennedy, 2002). It is especially helpful for continuous optimisation problems and draws inspiration from the social behaviour of fish and birds (Clerc and Kennedy, 2002).Below is the steps to be followed to implement the hybrid approach HPSOGA algorithm (Jatana N and Suri B, May 2020)(Kingma, D.P. and Ba, J, 2014). Begin by outlining the optimisation problem that has to be solved, together with the boundaries of the search space and the objective function. Organise probable solutions into a population known as "particles."

Every particle is a possible fix for the issue at hand.Initialise each particle's position and velocity within the search space of the issue at random. Based on the objective function, give each particle a fitness value. The problem-solving ability of each particle is gauged by its fitness value. Specify variables like the quantity of particles, maximum number of iterations, weight of inertia, acceleration coefficients (social and cognitive), and any other management factors.

The following actions are taken by the PSO algorithm for every iteration, or generation then adjust the Position and Velocity of the Particle in the population (Clerc and Kennedy, 2002)

Step 1: The population size P, acceleration constants c1 and c2, crossover probability Pc, mutation probability P m, partition number partno, number of variables in partition m, number of solutions in partition g, and maximum number of iterations Maxitr (***Jatana N and Suri B, May 2020)*** are among the parameters that the proposed HPSOGA algorithm sets first.

Step 2: After initializing the iteration counter t, each solution in the initial population is assessed and created at random.

Step 3: Until the termination criteria are met, the subsequent procedures are repeated.

Step 3.1: Using the typical particle swarm optimization technique (PSO) on the entire population, new solutions X ~t are produced.

Step 3.2: Use the GA selection operator to pick an intermediate population from the present one.

Step 3.3: The current population is divided into part no sub-populations, each of which has a size of X0~ðt\ size, or m g, where m is the number of variables in each partition and g is the number of solutions in each partition, in order to increase search diversity and solve the dimensionality issue.

Step 3.4 On every subpopulation, the arithmetical crossover operator is used.

Step 3.5 To prevent premature convergence, the genetic mutation operator is applied over the entire population.

Step 4 The population's elucidations are assessed through the computation of its fitness function. Until the termination criteria are met, the procedures are repeated overall with an increasing iteration number t.

Step 5 The optimal option is finally shown.

IV. Conclusion

Implement strong encryption mechanisms, such as WPA3, to secure data in transit. Avoid utilising vulnerable protocols such as WEP. There are possible future paths and unresolved issues in the physical layer security domain. This suggests that this field of study is still being researched and developing. The proposed approach is expected to solve possible future paths and unresolved issues in the physical layer security domain. It mainly addresses the concepts of secrecy capacity, feasible secrecy rate, and capacity-equivocation region, which are used to gauge the security of wireless communication.

References

[1] Choi, M. K., Robles, R. J., Hong, C. H., & Kim, T. H. (2008). Wireless network security: Vulnerabilities, threats and countermeasures. *International Journal of Multimedia and Ubiquitous Engineering*, *3*(3), 77-86.

[2] Choi, M. K., Robles, R. J., Hong, C. H., & Kim, T. H. (2008). Wireless network security: Vulnerabilities, threats and countermeasures. International Journal of Multimedia and Ubiquitous Engineering, 3(3), 77-86.

[3] Chhikara, P., & Patel, A. K. (2013). Enhancing network security using ant colony optimization. Global J. Comput. Sci. Technol. Netw. Web Secur, 13(4), 19-22.

[4] Dasgupta, D. (Ed.). (2012). *Artificial immune systems and their applications*. Springer Science & Business Media.

[5] How To Overcome Challenges For Remote Workers. https://hrchallenges.com/how-to-overcome-challenges-for-remote-workers/

[6] Jatana N and Suri B, "Particle swarm and genetic algorithm applied to mutation testing for test data generation: a comparative evaluation", Journal of King Saud University-Computer and Information Sciences, vol.32, no.4, pp.514-21, May 2020.

[7] Kaur, D. K. (2015). A ride from molecular recognition to development of optical sensors - An introduction. Kaav International Journal of Science, Engineering & Technology, 2(3), 28-50.

[8] Kingma, D.P. and Ba, J., "Adam: A method for stochastic optimization," arXiv preprint arXiv:1412.6980, 2014.

[9] Mirjalili, S., Mirjalili, S. M., & Lewis, A. (2014). Grey wolf optimizer. *Advances in engineering software*, *69*, 46-61.

[10] Noor, M. M., & Hassan, W. H. (2013). Wireless networks: developments, threats and countermeasures. *International Journal of Digital Information and Wireless Communications (IJDIWC)*, *3*(1), 125-140.

[11] Holland, J. H. (1992). Genetic algorithms. *Scientific american*, *267*(1), 66-73.

[12] P. (2017). Formal Verification of Energy Saving Techniques In Wireless Sensor Networks (WSN). Kaav International Journal of Science, Engineering & Technology, 4(3), 132-139.

[13] Potlapally, N. R., Ravi, S., Raghunathan, A., & Jha, N. K. (2005). A study of the energy consumption characteristics of cryptographic algorithms and security protocols. *IEEE Transactions on mobile computing*, *5*(2), 128-143.

[14] R., D. C. (2019). Energy Efficient Scheduling and Clustering in Wireless Sensor Networks: A Review. National Journal of Arts, Commerce & Scientific Research Review, 6(1), 274-279.

[15] Sathyavani, K. S., & Selvi, P. (2014). Wireless network security vulnerabilities, threats and countermeasures. In *International Conference on Information and Image Processing. Retrieved from http://www. conference. bonfring. org/papers/sankara_iciip2014/iciip89. pdf*.

[16] Teodorović, D. (2009). Bee colony optimization (BCO). In *Innovations in swarm intelligence* (pp. 39-60). Berlin, Heidelberg: Springer Berlin Heidelberg.

[17] Yang, X. S., & Deb, S. (2014). Cuckoo search: recent advances and applications. *Neural Computing and applications*, *24*, 169-174.

[18] Yang, X. S., & He, X. (2013). Bat algorithm: literature review and applications. *International Journal of Bio-inspired computation*, *5*(3), 141-149.

[19] Yang, X. S., & He, X. (2013). Firefly algorithm: recent advances and applications. *International journal of swarm intelligence*, *1*(1), 36-50.

[20] Wang, D., Tan, D. and Liu, L., "Particle swarm optimization algorithm: an overview," Soft Computing, vol.22, no.2, pp.387-408, 2018.