Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 32 No. 9s (2025)

Enhancing Network Security in Wireless Sensor Networks through Rsa-Based Secure Memory Management and Optimized Resource Allocation: A Comprehensive Survey

[1]C. Visalatchi, P, [2]Dr. K. S. Mohanasathiya, [3]C. Visalatchi,

[1]P. h.D Research Scholar, Department of Computer Science, VET Institute of Arts and Science (Co-education) College, Thindal, Erode, Tamil Nadu, India, visalatchic09@gmail.com.

[2]Assistant Professor and Research Supervisor, Department of Computer Science, VET Institute of Arts and Science (Co-education) College, Thindal, Erode, sathyaanandh08@gmail.com.

[3]Assistant Professor, Department of Computer Science, Kongu Arts and Science College (Autonomous), Erode, Tamil Nadu, India.

Article History:

Received: 12-01-2025

Revised: 15-02-2025

Accepted: 01-03-2025

Abstract:

To improve the safety of networks in Wireless Sensor Networks (WSNs), this study offers a thorough analysis and presents a unique architecture that combines the best utilization of resources with RSA-based secure control of memory. WSNs are essential for many tasks, including weapons systems and tracking the environment, but because of their centralized architecture and shortages of resources, they are very susceptible to breaches of security. This assessment provides a critical analysis of current security mechanisms, emphasizing resource distribution tactics, memory administration approaches, and methods of encryption particularly designed for wireless sensor networks. Building on these discoveries, the study suggests a novel paradigm for the secure utilization of memory in sensor networks that makes use of the RSA encryption method to protect information security and secrecy. While upholding stringent safety criteria, the structure uses effective utilization of resources approaches to increase WSNs' overall effectiveness and efficacy. The proposed method provides a strong solution that improves operational effectiveness and security while addressing the particular difficulties presented by WSNs.

Keywords: Network Security; RSA Encryption; Secure Memory Management; Optimized Resource Allocation; Cyber security; Data Integrity; Encryption Techniques; System Efficiency; Wireless Sensor Networks

1. Introduction

Since its introduction to the public in 1977 with the introduction of public-key encoding and DES, encryption has played a significant role in safe computation, initially in the DoD and other national agencies. The necessity to safeguard personal information grew along with the public's usage of computers. This protocol combines public and private-key encryption. More recently, commodity machines have included Full Disk Encryption (FDE) ensures the secrecy of any information saved on disk [1]. Several commercially feasible FDE implementations have been made possible by recent advancements in hardware-based encryption and processor performancecan be attributed to Moore's

law. TrueCrypt, Bitlocker, and PGPDisk are software solutions for FDE. The use of FDE technology is growing as an outcome of multiple causes. The emergence of mobile computing and the extensive exchange of knowledge over the World Wide Web has sparked worries about access to information in the real world. A lot of information intrusions have been made public, which has increased knowledge of risks [2].

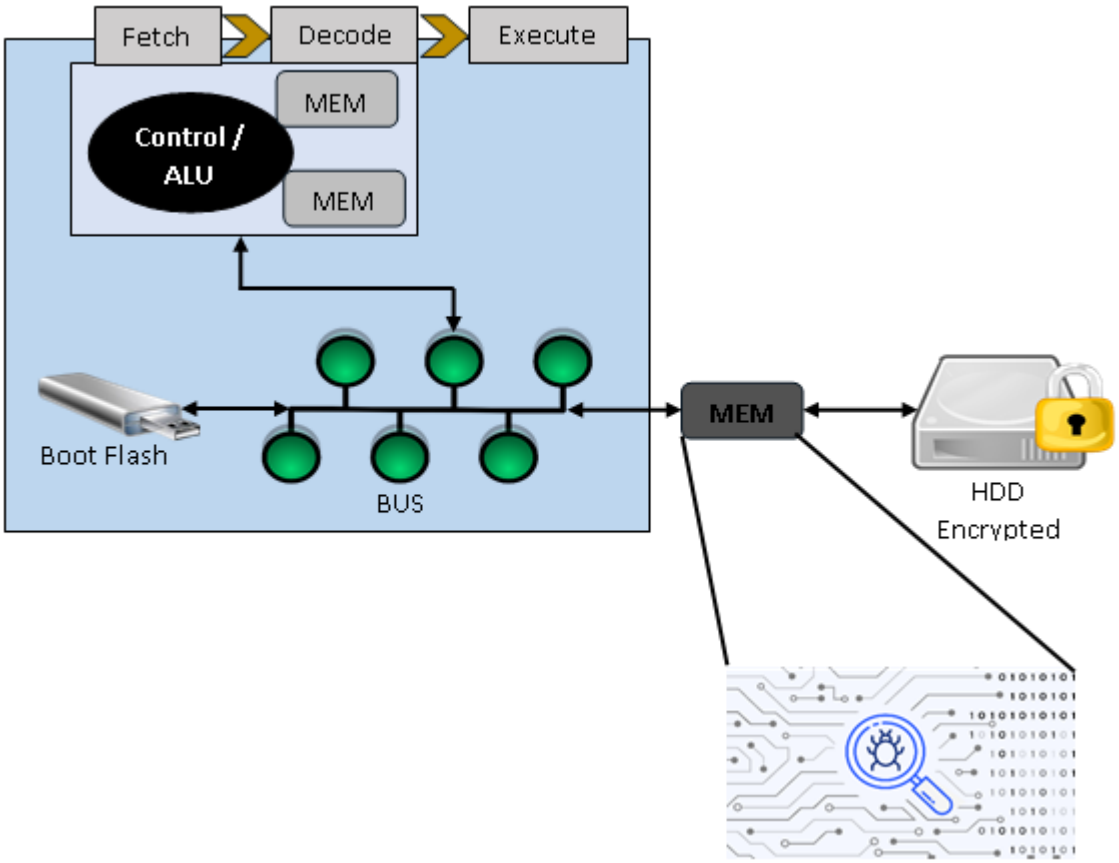

Figure1. Architecture of code and data full disk encryption

Figure 1 illustrates a significant flaw in systems that exists even with FDE: information and programs kept in memory are unsecured, or exposed in the visible. It has been demonstrated that even apps created with safety in mind are susceptible. For instance, when a key has been used, cryptographic libraries are intended to zeroize it or overwrite it with zeros, preventing access to the key [3]. Compiler optimization occasionally eliminates this zeroing of code because it seems unnecessary, recreating the vulnerability. Some data remains in RAM for many minutes even in the absence of cooling. On the other hand, cooling lowers recovering mistakes by slowing down the rate of losing information. Certain methods such as DMA-firewire assaultavoid FDE to facilitate forensic examination.Both legitimate authorities terrorist organizations and other adversaries have identical access to these tactics [4]. Data may be intercepted or injected over the bus lines connecting system elements using an especially successful technique called bus-snooping and injection. The Xbox gaming machine has been compromised through the use of this exploitation technique. Keys were intercepted during their transfer between the CPU and read-only memory using bus eavesdropping [5]. The device used for gaming may be utilized for illegal purposes by using the same processors to power various OS on it. An attacker has a brief window of opportunity to retrieve crucial information

between power cycles while accessing data in typical dynamic RAM. Nevertheless, emerging nonvolatile alternatives such as ferroelectric, magnetic, and flash memory are supplementing or replacing dynamic RAM. These alternatives provide several advantages like power failure tolerance and energy conservation [6]. While the "ready boost" function in Windows 7 and Vista uses flash storage to supplement standard RAM, the other two kinds of memory have an opportunity to substitute RAM. These nonvolatile memories let data and assaults endure forever. It's significant to note that Microsoft created the prepared enhanced functionality to encrypt all flash information in anticipation of safety hazards related to persistent memories, making it more challenging for forensic analysts to access significant information [7].

Remembering encoding improves system safety, but attacks on gadgets can still occur by using acid to etch separate the chip wallsexposing inner bus lines for microprobing, or by using opposite side communicates like electromagnetic and energy evaluations. Cache attacks can affect software-based encryption protocols that use key augmentation databases such as Advanced Encryption Standard (AES); an attacker's program keeps track of and timing cache requests [8]. All of these assaults usually aim for the secret key for encryption that is concealed inside the chip border. Except for cache attacks, the majority of these techniques need specialized expertise, greatly increase the attacker's effort, and cannot be automatically exploited via an internet connection. Both lawbreakers and well-meaning users have equal access to protections like FDE. Data about illegal behavior has been protected using encrypted disks to thwart successful prosecutions [9].

Memories encrypting might be used to further secure illicit activities and frustrate some of the tactics revealed to help authorities. Machines are used nowadays in a wide range of information processing programs, from routine to delicate. These applications can be offline or online and protect consumers making online payments through e-commerce or businesses sharing confidential information on the Internet [10]. To protect information that depends on cryptographic methods such as symmetric and asymmetric cryptography, integrity of information procedures, and identification procedures, several safety techniques have been developed. The term "safety" refers to the set of qualities that include accessibility (which prevents unapproved withholding of data), integrity (which prevents unauthorized modification or deletion of data), and privacy (which prevents unauthorized release of data). OSI provides the generic safety concept that guarantees the privacy of data throughout transmission [11].

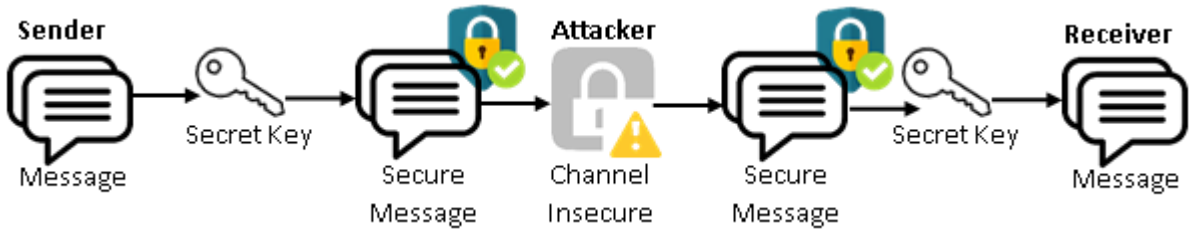

Figure 2: Generalized Security model of data

By transforming ordinary text into hidden knowledge and vice versa using a safety growth system (cryptographic algorithm) and transmitting through a channel that is insecure, the previously

mentioned approach, as illustrated in Figure 2, ensures the safety of data between both senders and recipients during network interaction. This prevents adversaries (attackers) from attacking [12].

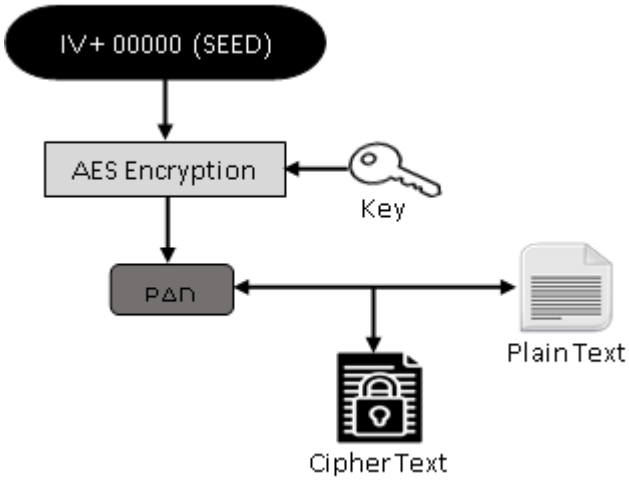

Figure 3. Encryption (One time pad)

The pad is created by selectively Orion (XORing) the plain language and the seed information using a distinct key, resulting in the text used for the cipher shown in Figure 3. Since counter confidentiality is not necessary, the counter in storage encryption devices is either saved within the organization in a caching table that translates to a storage location, or unprotected within the protected storage directly (i.e., RAM) [13]. In the event of internal navigation, the pad is recreated entirely with the information that has been encrypted, producing what was originally plain text, using the counter (and possibly some other element, like the virtual addressing) and activation vectors. This generation may overlap with the storage read since the procedure for encryption is no longer dependent on the information in memorieslessens the effect of decoding on efficiency [14].

1.1 Problem Statement

In many vital applications, including healthcare, military operations, and environmental surveillance, WSNs are indispensable for the safe and effective transfer of information. Constrained compute capacity and power limitations, WSNs are intrinsically susceptible to safety hazards. Conventional safety measures frequently fall short of providing sufficient defense without impairing network functionality, which can result in problems including breaches of information, illegal access, and wasteful resource use.

1.2 Motivation

The increasing use of WSNs in delicate and important programs, where transmission of information safety and effectiveness are crucial, is the driving force behind this study. WSNs are more vulnerable to safety incidents like information surveillance, unauthorized entry, and denial-of-service assaults are implemented in more varied and restricted in resource situations.While successful in some situations, conventional safety techniques frequently fall short in WSNs because of their constrained computation and electrical power, resulting in a trade-off between safety and network efficiency. The demand for creative solutions is driven by the difficulty of maintaining adequate safety without

sacrificing network performance. RSA encryption, which is well-known for its robust data security properties, presents a potential method; yet, the intrinsic restrictions of WSNs prevent their widespread use. The present study aims to solve these issues by creating an architecture that combines the best utilization of resources with RSA-based safe handling of memoryoffering a complete solution that improves both safety and effectiveness in WSNs. The driving force is the possibility of greatly enhancing WSNs' robustness and dependability, guaranteeing their capacity to function well in progressively harsher and resource-constrained circumstances.

2. Cryptographic Initiatives

In the age of electronic devices, achieving the security of data necessitates a wide range of scientific and legal expertise. Although cryptography offers the technological tools to do it, there is no certainty that all of the required information safety goals can be satisfied [15]. The investigation of computational techniques for protecting information, including entity authorization, data confidentiality and secrecy, and information origins authorization, is known as cryptography. Security of data may be provided by a variety of methods, not just cryptography. Symmetric and asymmetric cryptography are the two main categories of cryptography [16].

2.1 Symmetric and Asymmetric Cryptography

Figure 4 illustrates how a symmetric encryption systemcomprises five elementsand use the key for both encryption and decryption

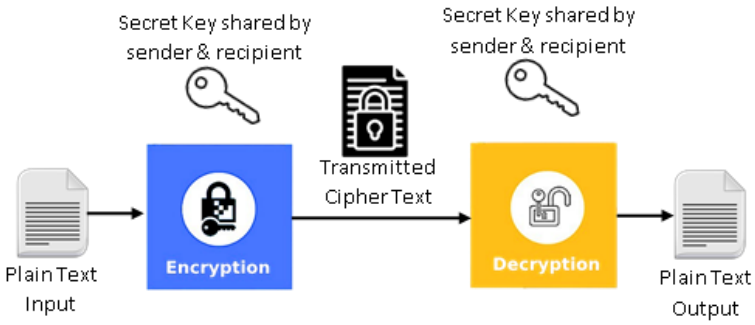

Figure 4: Symmetric Cryptography Model

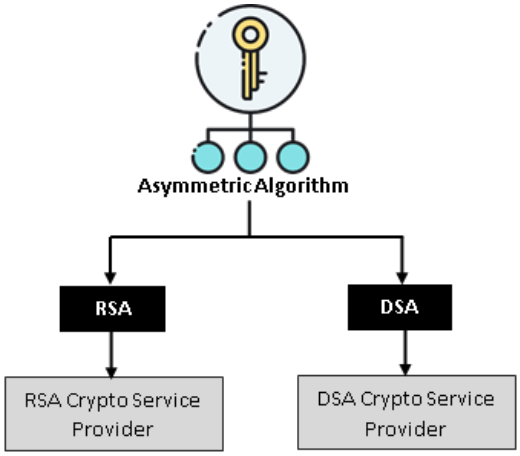

Figure 5: Asymmetric Cryptography Model

Using a private key for handling the communication, an electronic signature is created during verification. To validate the signed document, the signer's matching public key is applied to the signature's significance, and the outcome is compared with the contents of the email. A communication's digest or hash is often encrypted and used as a seal of authenticity. The digital signatures can be obtained by certain public key methods, while others offer confidentiality and dissemination of keys [17]. The organizational structure of asymmetric method classes is depicted in Figure 5.

2.2 RSA Messaging (Efficient and Lightweight)Scheme

RSA is a public key system that utilizes exponential growth across numbers in an infinite field. Every individual creates a public-private key pair for an RSA key configuration by choosing two big prime numbers at random. The personal key used for decoding is kept secret from others, while the public key used for encrypting is made public. For exponential computation, it applies the squared and multiplier procedure and Euler's theorem. To calculate the outcome, the base amount is repeatedly squared, and the coefficients are then combined [18]. Computational temporal, and brute force key search attacks are all directed at the RSA method. By introducing random delays and utilizing a consistent exponentiation time, the assaults may be stopped. The RSA method's properties include confidentiality and privacy, honesty, authorization, and non-repudiation. In LAN-to-LAN and VPN scenarios, true E2E encryption is required. The plan is now available as accessible programs [19]. Installing the messaging program is a must for both clients and peers. The person using them only has to click on the programs to launch the communication system after deployment is completed. Before SHRSA individual communications, the four-layered stacked type identification is operational. To further securely ensure each peer's authenticity, we have implemented these additional 4 stages of verification on top of SHRSA's primary identification shown in Figure 6. The RSA variations mentioned above, which serve as the foundation for the SHRSA communication technique, each have their one-layer identification [20].

2.2.1 Encryption

The encryption strategy that is employed defines the process of encoding a communication. The message or data intended to be transferred is given in an unencrypted format when using an encryption technique. After that, the message as it is written is encrypted using a technique for encryption to produce ciphertextneeds to be decoded to be understood. Every block that makes up the encrypted message has a value smaller than the standard module n [21]. The algorithm for encryption is provided by

$$C = M_e \ mod \ n \quad (1)$$

Where:M - Block of plain text; C - Block of cipher text; e - Public key

2.2.2 Decryption

It is the procedure of decoding ciphertext to produce the message as plain text in an understandable manner. The algorithm created by RSA limits the ability to decode data to those having access to the confidential key [22]. The following is the decoding method:

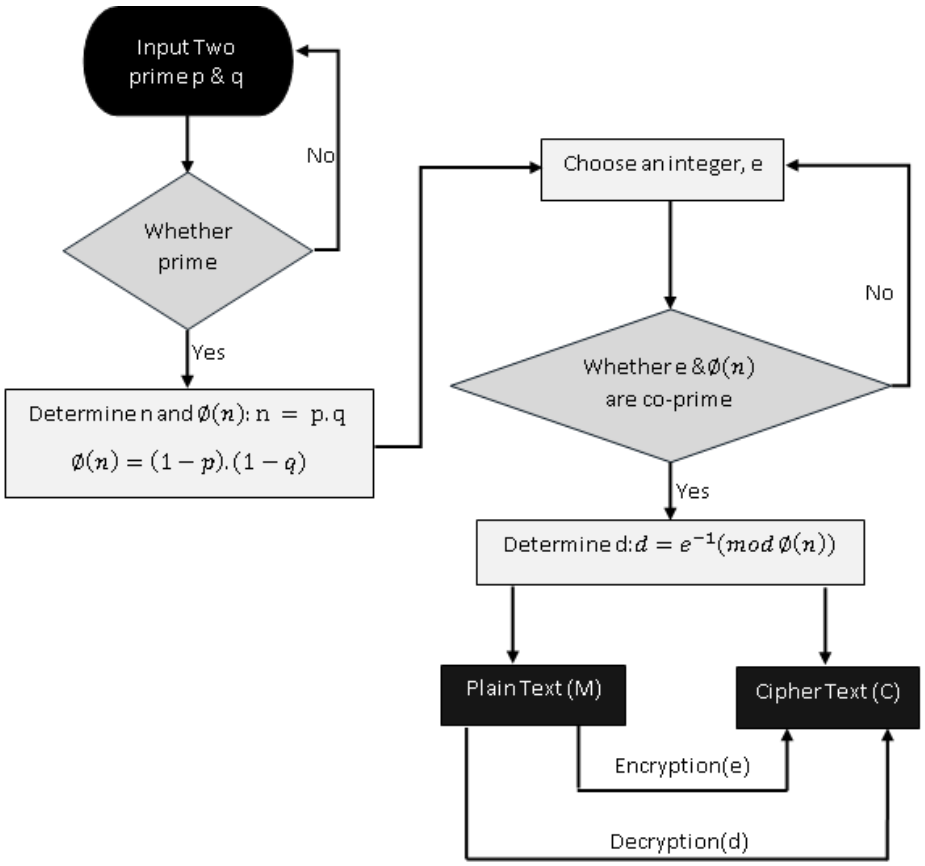

Figure 6: RSA method for encryption and decryption

$M = C^d \, mod \, n$ (2)

Where:M - Block of plain text; C-Block of cipher text; d- Private key

2.2.3 RSA algorithm limitations

This includes the computational cost due to the requirement for two separate keys and the speed issue, as previously outlined. Another potential safety concern is the loss of the private key. Since RSA uses a private key, it has been criticized for this vulnerability [23]. According to [24], the private key is used during the decryption process; therefore, if an unauthorized person gains access to the private key, the entire security of the RSA algorithm is compromised. RSA is susceptible to mathematical factoring attacks [25], necessitating improvements. The execution time of the RSA method was found to be longer compared to other techniques. It was discovered that RSA was relatively easy to factor, as the modulus n used was simply the product of two prime numbers, making it easier to retrieve the original encrypted information. Identified several weaknesses in the RSA method and enhanced the model by improving privacy and eliminating unnecessary information using the K-nearest neighbors approach [26].Table 1 summarizes recent research on lightweight and efficient RSA.

Table 1: Related Works based on RSA

Title	Year	Techniques	Key Features	Advantages	Limitations
"A Lightweight RSA Encryption Scheme for Resource-Constrained Devices" [27]	2024	Optimized RSA algorithms	Focuses on reducing computational overhead and energy consumption for lot devices.	Efficient for low-power devices. reduces energy usage.	May not offer the highest security levels.
"Efficient RSA-Based Messaging for Mobile and Embedded Systems" [28]	2023	RSA with Modular Arithmetic Optimization	Enhancements in RSA operations to improve performance on mobile and embedded systems.	Improves performance and reduces latency in resource-limited environments.	Limited applicability to non-embedded systems.
"Design and Implementation of a Lightweight RSA Protocol for Secure Communication in Ad Hoc Networks" [29]	2023	Lightweight RSA with Key Compression	Tailored for ad hoc networks with reduced key sizes and computational requirements.	Enhances security while maintaining efficiency in ad hoc settings.	Potential trade-offs in key security and scalability.
"Optimizing RSA for Low-Power Wireless Sensor Networks" [30]	2022	RSA with Enhanced Padding Schemes	Introduces efficient padding techniques to reduce computational load in sensor networks.	Reduces encryption and decryption time, suited for low-power sensors.	May require adaptation to different network topologies.
"RSA-Based Secure Messaging with Energy Efficiency for	2022	Energy-Aware RSA Encryption	Focuses on balancing security and energy efficiency	Enhances energy efficiency while providing	Energy optimizations may affect encryption speed.

loT Applications" [31]			specifically for loT applications.	robust security.	
"Lightweight RSA Implementation for Secure Messaging in Wearable Devices" [32]	2021	Hardware-Accelerated RSA	Utilizes hardware acceleration to achieve lightweight RSA encryption for wearable devices.	Improves performance and security for wearable technology.	Hardware dependencies might limit flexibility.
"Efficient RSA Key Management and Encryption for Mobile Platforms" [33]	2021	RSA with Reduced Key Sizes and Accelerated Computations	Optimized for mobile platforms with reduced key sizes and faster computations.	Enhances performance and reduces overhead on mobile devices.	Reduced key sizes may impact security.
"Scalable and Lightweight RSA Encryption for Networked Embedded Systems" [34]	2020	Scalable RSA with Optimization Techniques	Focuses on scalability and efficiency for networked embedded systems.	Offers scalable solutions suitable for large networks.	Scalability improvements may introduce complexity.

The main methods, characteristics, benefits, and drawbacks of each study are highlighted in this table, which provides an overview of existing research on effective and lightweight RSA messaging systems.Using the Trusted Third-Party [TTP]approach, challenge class with an IP address that is distinctive to validate the peer. Although the structure of the SHRSA system is set up to allow for SHRSA customers, any additional SHRSA customers wishing to communicate must wait for P2P verification [35]. As a result, although they can connect to the SHRSA server, they are unable to start a message. In this manner, the structure guarantees pure P2P encrypted communication before the initiation of the primary SHRSA communication method. Everyone uses a three-way handshaking protocol between peers in layer 2 of the four-layered verification scheme. For layer 3 of the 4-layer authenticating stack, we are employing the primary Diffie-Hellman key transfer mechanism. The PFS is being used here because of its special property benefit. By protecting the keys in transit, the

DH-PFS-based 4-layered authenticating stack prevents identification transfer to passive eavesdropping attacks at the network interface [36]. This four-layered stacked structure eliminates the change of password release with four times real-time key negotiation. Since IPsec is required by the IPv6 protocol, it comes pre-installed on all IPv6-capable computers and may be enabled at any time. Everyone is employing the NetBeans IDE 8.2 and has created the SHRSA communications mechanism in Java. There are 42 classes total for SHRSA both decryption and encryption, with 21 classes for every [37]. Demonstrated that only one peer at a time can initiate safe E2E-authorized communications; all other peers will remain linked to one another but will need to wait for the existing peer in communication to conclude before proceeding. Requests have built an API for RSA clients and server communications for evaluation and contrast purposes. Currently developing nine-layered safe communication communication stack APIs for the SHRSA client and server components [38].

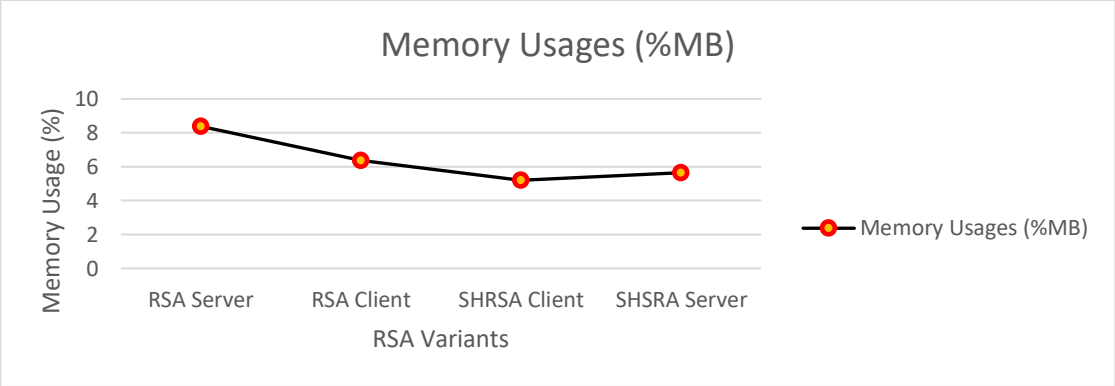

Figure 7: Comparison of memory usage (Server and Client)

512 MB of RAM was allotted to us for assessment. After running each of these four APIs five times, we're going to average the results after two minutes of continuous communication in both directions. The five-times running median of storage occupancy is shown in Figure 7. Itdisplays the storage occupancy information that has been obtained. In comparison to the primary RSA encryption, the server and client APIs of the SHRSA communication method use 1%–3% less RAM during communication [39].

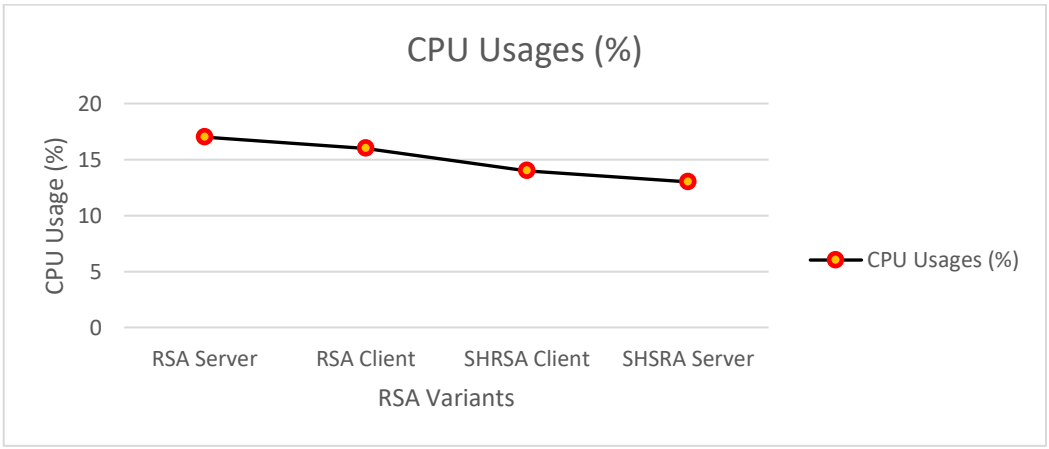

Figure 8: Comparison of CPU usage (Server and Client)

SHRSA client and server processes for two minutes throughout this continual communication assessment period. After running each of these four APIs five times, averaged the results. Continuously messaged the other way for two minutes. SHRSA servers and clients use a lot less CPU power than the primary RSA instant messaging client and server. Figure 8 displays the CPU occupancy data obtained. Overall, the SHRSA encryption uses 2%–4% less CPU power than the primary RSA cipher [40].

3. Multiprocessor Enhancement

For monolithic manufacturers, the main problem is the effectiveness of memory-to-cache secrecy; however, systems with multiple processors also need to safeguard cache-to-cache traffic. Coordination of communications across processors in SMPs can be achieved through the shared bus connecting caches and memory. DSM systems have to employ the transmission of messages instead of this sharing shown in Figure 9 [41]. DSM systems are easier to see than monolithic processors to visible connecting lines at the rear of server racks. After every reboot, a physical device in the processor makes sure that the order of numbers starts over at a new place. In addition to the standard crypto processors integrated into each CPU core, the north bridge memories controllers feature an independent crypto-unit for memory-to-cache transactions. By storing 64-bit number sequences in RAM for these promotions, 25% less space is accessible.

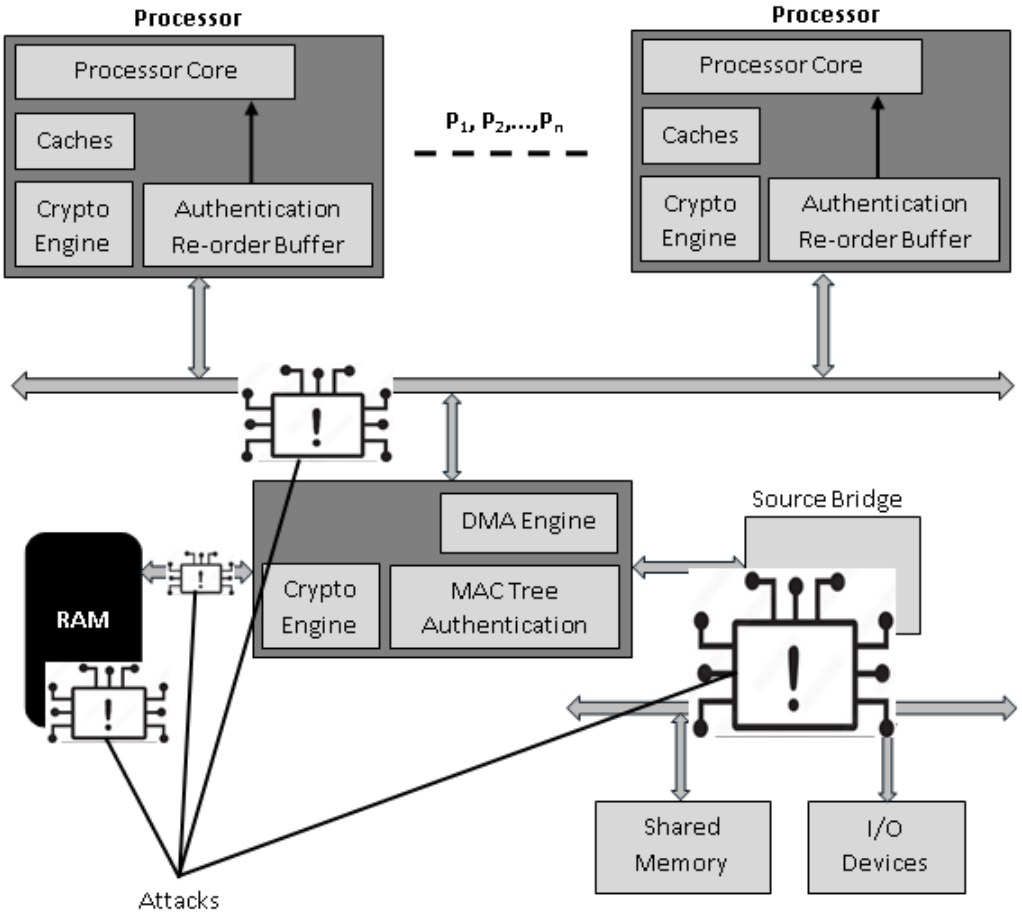

Figure 9:Architectutre of SMP (Encryption of Memory)

The segments are then chained and linked so that the output of one block is encrypted before it is encrypted by XORing its input with the subsequent one. A sequence of access is implied by CBC since every block is dependent upon every other block. Since only one previously encoded block needs to be kept at every processor (i.e., having access to previously encoded blocks is not necessary). Storage for a group data table and grouping processing matrices are included in this memory [42]. Every SHU uses the collective processing matrix to decide whether to read the broadcast messages. The matrix, presuming an aggregate of 32 processors, is just 640 bytes in size. The metadata table is anticipated to be 149KB in size, holds the secret data needed for group communication, such as the symmetrical keys and circuits. Eleven more bus lines are utilized for group ID information and control signals shown in Figure 9. The SENSS system is enhanced by utilizing Galois Counter-Mode (GCM) AES, which concurrently offers cryptography and verification [43].

The processors all receive the blocks of counters as well, enabling them to start pre-calculating pads. A keystream (pad) queue, keystream pool, and keystream cache are present in every CPU. There are pads for encryption in both the waiting area and the cache. While the cache holds previously employed pads, the waiting room is filled with brand-new pads. The researchers assert that since more than 25% of pads are utilized again and again, their technique grows well to high amounts of processing and that pads might be recycled as long as the encryption has not been altered. The broadcast strategy is used to choose the pads for incoming information from the keystream pool, which stores the pads determined by predictability [44]. Three approaches—private, shared, and caching counting streamare proposed by the researchers for controlling the pad of counters. Table 2 are maintained within every processor in the first secret approach, with distinct counters for sending and receiving activities when communicating with any other process in the overall system. Despite enabling very flawless pad hit rates and hence little overhead, this method has significant storage requirements (180KB per processor for a 1,024-processor DSM). By removing half of the surface area, the additional shared method seeks to decrease the amount of storage needed. For sending pads, a single counter is maintained rather than a record of send counters for every microprocessor. Because communications are less likely to be delivered simultaneously and must consequently be recomputed, this increases implementation complexity [45].

Table 2:Related Works based on RSA

Title	Year	Techniques	Key Features	Advantages	Limitations
"A Survey of Memory Encryption Techniques" [46]	2023	AES, RSA ECC	Overview of traditional encryption algorithms applied to memory.	Comprehensive comparison covers multiple methods	May lack depth in newer, less common methods

"Enhancing Memory Security with Advanced [Encryption Methods" [47]	2023	AES-GCM. ChaCha20	Focus on performance and security improvements in modern encryption methods.	Improved performance an security over traditional methods	Limited discussion on implementation challenges.
"High-Efficiency Memory Encryption for Cloud Computing" [48]	2022	AES HMAC. Key Wrapping	Techniques optimized for cloud environments including key management strategies.	Tailored for cloud environments enhances security	May not address all types of cloud architectures
"Hardware-Assisted Memory Encryption Techniques: A Survey" [49]	2022	Intel SGX. AMD SEV	Reviews hardware-based memory encryption solutions and their implementations	Leveraging hardware feature, high security	Limited to hardware-assisted solutions, not software-based
"Comparative Study of Memory Encryption Algorithms for IoT Devices" [50]	2022	AES TinyJAMBU	Evaluation of encryption algorithms specifically for resource-constrained loT devices	Focused on IoT constraints, efficient algorithms	May not apply to more powerful devices.
"Memory Encryption for Secure Virtualization Environments" [51]	2021	AES, VM Encryption	Addresses memory encryption in virtualized environments. including hypervisor considerations.	Relevant to virtualization, covers hypervisor aspects	Specific to virtualized systems, may not generalize
"Efficient Memory Encryption Strategies for Embedded Systems" [52]	2021	AES, TEA	Tailored encryption strategies for embedded systems with resource	Optimized for embedded systems, good performance.	May not be suitable for high-performance systems.

			constraints.		
"Memory Encryption for Enhanced Data Privacy: A Review" [53]	2021	AES-CCM, RSA	Focuses on data privacy improvements through encryption techniques.	Emphasis on privacy, covers various methods	Privacy-centric, may not address all security aspects.

Table 2 gives a summary of the most important methods, salient characteristics, benefits, and drawbacks of each work in the current literature on memory encrypting approaches.

3.1 Memory Encryption

Since multiple processors have distinct memory utilization patterns (pads are updated solely on storage transfers), the pads in their local caching can grow inaccurate when every processor executes rapid data encryption on its memory. Processors A and B both have information D cached and that D's pad is constant throughout both processors' pads caching. Afterward, B won't be able to use the local old pad for D if A moves D down to the RAM and upgrades D's pad. There are similarities between this issue and the classic cache consistency issue. Therefore, to manage the altering of a pad, species may either utilize a "write invalidated" or a "write update" method. When a pad is modified in one cache under "write invalidate," the other cache versions of the pad are rendered invalid via an invalidated communication delivered across the bus. Query information is issued on the bus to obtain the most recent duplicate if the pad needs to be utilized by another processor in the future.

3.2 Memory integrity check

Each CPU keeps its local portion of the hash tree while memory identification is active, and the tree is partially cached locally. As previously mentioned by the pads, it is simple to see how tree node inconsistencies across various processors might occur. We may use a similar approach to implement the "write invalidate" or "write update" protocols for recently updated information and hashes. The complexity comes from the possibility of receiving numerous invalidated or updated signals.It is possible that the parent node in the hashing tree might not remain in the cache for hash upgrading when a cache line is removed to storage. Either protocol can update the hash value coherently if the underlying node is in a different caching; in this case, the procedure ends. To authenticate with a parent node and modify its hash value in the grandparent's nodes, if the parental node is in the recall, species must both retrieve it and start having an additional connection to the grandparent node. The consistency of the changed grandparent node must be managed once again. Until a node is reached in the local cache, where no more associated nodes are required to be informed this process is repeated.

4. WSN Security

Ad hoc and networked sensors require safety as an add-on for route and packet forwarding. The nodes in ad-hoc and networked sensors carry out the basic functions. When these transmitters go in and out of transmission assortment, it constantly varies. An ad-hoc network's locations have an a priori trust connection, and entity identification may be adequate to guarantee the proper execution of

crucial network operations.To lessen security hazards, a safety mechanism that works effectively with many networks should be employed. Cryptography provides a solution to this issue by strengthening the safety of information and offering the fundamentals of safe communication and e-commerce.

A low-power group key administration technique based on LKH++ was described. To offer safety in WSN. Building a secure tree was the method used to handle the group keys within the network. The proposed plan described how to create trees and how to keep the keys. The network's reliability and safety were improved by the wireless LKH. The outcomes demonstrated that compute overhead, key capacity for storage, and security had all decreased. In homogenous WSN, have proposed a crucial pre-distribution strategy for grouping deployments. The proposed plan took advantage of partial position information to increase network durability and connectedness. For connection, it took into account both the physical graph and the block graph. The group distribution improved network connection without increasing storage requirements or compromising resilience, according to the outcomes. To lessen threats of collusion in WSNs, developed improved access polynomials-based self-healing key administration systems that made use of broadcast authenticating. To break the access polynomials' safety, two approaches were presented. The entire session frequency was used to define the colluding sensitivity. The use of the window that slides and adjusts accessing polynomials allowed for the achievement of safety and the tolerance of lost packets. Errors in accessing polynomials have been avoided and the usage of resources has decreased.

The findings demonstrated that the proposed strategy prevented the collusion approach and achieved both forward and reverse confidentiality. 2-D backward chain of keys technique-based continuous secured solution for static heterogeneity WSN. Using m-dimensional backward key chains, several linked and disjoint key sources were produced. The BS was pre-distributed with all 44 master keys and the pool's keys under the terms of the key pre-distribution procedure. The route key generation approach was used in cases where direct key development failed. A Hierarchical Key Management System (HKMS) was introduced to enhance safety in clustering WSNs. The keys were distributed based on hop counts and a one-way function to minimize overhead. The collective keys were utilized by the CH and its constituents, whereas the session-specific keys were dispersed to the nodes that collected data. The keys to the malicious nodes can be revealed by the hacked nodes.

As a result, several keys based on different hop counts were established to safeguard the network against the hacked nodes. The produced keys varied over time and were only utilized for a specific amount of time. A grouping-based key administration strategy was presented to address problems with scalability, inefficiency, and communication overhead inside the WSN clusters. The cluster heads created the group keys, which the subgroup heads then dispersed over the safe channels. The group key generation process was significantly enhanced by the grouping. The findings demonstrated that the recommended plan extended the system's total lifetime by consuming less energy and time. To handle the keys in WSN, indicated a group authorization mechanism that is scalable, adaptive, and based on cryptography. The administrator has been granted permission for access to achieve versatility in any kind of network framework, including peer-to-peer and hierarchy networks. By creating a new secure fitting operation, the ability to scale was achieved.

The hash code computations and security filter functions were used to update the group's membership keys. The technique based on cryptography was more effective than the other methods. To maintain group keys in WSN, presented an improved identity-based use of cryptography or EIBC. Publicly known identification was made possible and the sharing of certificates containing public keys was discontinued. Multicast group key administration mathematical framework was developed. The outcomes demonstrated how incredibly secure the EIBC system was. To safeguard the data in the WSN, proposed an energy-efficient key administration system. Three keys have been created using the proposed method: an ensemble key, a private key, and a pairing key. While every node shared the collective key, the BSs and neighbor nodes shared each of the individual and individual keys, correspondingly. The polynomial equation was utilized to compute the keys during the beginning, participation change, and key compromise processes.

The proposed system achieves reduced determine, interpersonal interaction, and storage expenditures as compared with the current techniques. A group key management strategy (KIEGKMS) has been proposed by integrating the together key pre-distribution in WSN. As nodes were added to or removed from the connection, the collective keys were promptly changed. Key Insulated Encryption (KIE) was used to encrypt the keys. Only the legitimate nodes received the encryption keys. The outcomes demonstrated that both backward and forward safety were accomplished by the KIE-GKMS technique. An Elliptic Curve Cryptography (ECC)--based mutual verification and key administration strategy for WSNs was described. In a specific session, a new timestamp technique was used between two mutual nodes. Asymmetric key cryptography provided mobility, while reciprocal authentication improved safety. With limited storage requirements, safe interaction was accomplished with the ECC-based system. To determine which key systems for administration work best for specific programs, examined several WSN techniques. This survey covered the characteristics and requirements of several key management systems. Framework for key management strategies was created to select the right schemes for a given WSN environment.

4.1 Research Gap

The following study needs may be found in the context of improving network security in WSNs through RSA-based safe memory administration and the optimum resource allocations:

- RSA Integration with Storage Leadership: Despite being widely used in data encryption, little is known about how RSA integrates with safe information management techniques in wireless sensor networks (WSNs). To solve safety and effectiveness issues in WSNs, development is required to create architectures that integrate sophisticated memory management strategies with RSA-based encryption.
- Efficiency cost: In WSNs with limited resources, RSA encryption, which is renowned for its complexity of computation, might result in a large efficiency cost. Investigation is required to determine how to best optimize RSA implementations to reduce this cost and preserve excellent safety, especially in low-power and low-bandwidth settings.
- Resource Allocation Strategies: In WSNs, balancing consumption of energy, power consumption, and memory utilization requires efficient allocation of resources. To ensure that resource management tactics are optimized and that encryption and safety features do not negatively

impact the general effectiveness and lifetime of networks, research is required to create and assess RSA-based systems.

- Adaptive Security Mechanisms: WSNs frequently function in unstable, dynamic situations where nodes' capabilities might fluctuate and the state of the overall network can alter. Developing adaptive RSA-based security systems that can dynamically adapt to these changing settings and offer adequate safety without sacrificing effectiveness is lacking.
- RSA-based Solutions' Scalability: As WSNs get bigger, it gets harder to keep RSA-based encryption and memory management working effectively and efficiently. Scalability concerns must be investigated to guarantee that RSA-based systems can withstand extensive WSN deployments without suffering appreciable performance loss.
- Impact of Emerging Threats: New threats constantly appearing, and the safety environment is always changing. Investigation is required to determine how well RSA-based safety measures work against new and sophisticated attacks as well as to create flexible defenses against these changing threats.
- Real-World Implementation and Usability: The majority of studies to date have been conducted in controlled situations or theoretical models. To verify theoretical results and tackle practical issues, empirical research and real-world applications of RSA-based secured memory administration and allocation of resources in WSNs are required.
- Trade-offs between Safety and Power Effectiveness: The resource-intensive nature of the encryption algorithm RSA may force trade-offs between safety and power economy. It is necessary to do a study to determine how to strike a balance between these trade-offs so that improvements to security don't negatively affect WSN energy consumption too much.

By combining the benefits of RSA encryption technology with appropriate memory administration and allocation of resources strategies, filling up these gaps in knowledge can result in more workable and feasible approaches to improve security measures in WSNs.

5. Conclusions

In conclusion, improving the safety of networks in WSN via efficient resource allocation and RSA-based secured control of memory is a promising but difficult field of study. Although the encryption method RSA offers strong security, further research is needed to determine how to integrate it with safe memory administration and effective allocation of resources in wireless sensor networks. According to recent studies, RSA implementations must be optimized to reduce efficiency expenditures and uphold strict safety criteria. The practical implementation also requires addressing flexibility, adjusting to changing network circumstances, and striking a balance between conserving energy and safety. Future work should concentrate on creating adaptive RSA-based solutions that are practical, address new threats, and blend in smoothly with contemporary storage management strategies. Create more flexible, safe, and effective network setting that satisfies the stringent needs of modern applications for WSNs by filling up these gaps in knowledge.

References

1. Anusuya Devi, V., &Sampradeepraj, T. (2024). End-to-End Self-organizing Intelligent Security Model for Wireless Sensor Network based on a Hybrid (AES–RSA) Cryptography. *Wireless Personal Communications*, 1-29.
2. Qasem, M. A., Thabit, F., Can, O., Naji, E., Alkhzaimi, H. A., Patil, P. R., &Thorat, S. B. (2024). Cryptography algorithms for improving the security of cloud-based internet of things. *Security and Privacy*, *7*(4), e378.
3. Popoola, O., Rodrigues, M. A., Marchang, J., Shenfield, A., Ikpehai, A., &Popoola, J. (2024). An optimized hybrid encryption framework for smart home healthcare: Ensuring data confidentiality and security. *Internet of Things*, *27*, 101314.
4. Hasan, M. K., Weichen, Z., Safie, N., Ahmed, F. R. A., & Ghazal, T. M. (2024). A Survey on Key Agreement and Authentication Protocol for Internet of Things Application. *IEEE Access*.
5. Valluri, B. P., & Sharma, N. (2024). Exceptional key based node validation for secure data transmission using asymmetric cryptography in wireless sensor networks. *Measurement: Sensors*, *33*, 101150.
6. Heo, J. W., Ramachandran, G. S., Dorri, A., &Jurdak, R. (2024). Blockchain data storage optimisations: a comprehensive survey. *ACM Computing Surveys*, *56*(7), 1-27.
7. Tomović, S., Krivokapić, B., Nađ, Đ., &Radusinović, I. (2024). BEKMP: A Blockchain-Enabled Key Management Protocol for Underwater Acoustic Sensor Networks. *IEEE Access*.
8. Asiri, M. M., Alfraihi, H., Said, Y., Othman, K. M., Salama, A. S., &Marzouk, R. (2024). Securing Consumer Electronics Devices: A Blockchain-Based Access Management Approach Enhanced by Deep Learning Threat Modeling for IoT Ecosystems. *IEEE Access*.
9. Ali, S., &Anwer, F. (2024). Secure IoT framework for authentication and confidentiality using hybrid cryptographic schemes. *International Journal of Information Technology*, *16*(4), 2053-2067.
10. Hossain, M., Kayas, G., Hasan, R., Skjellum, A., Noor, S., & Islam, S. R. (2024). A Holistic Analysis of Internet of Things (IoT) Security: Principles, Practices, and New Perspectives. *Future Internet*, *16*(2), 40.
11. Supriya K, S., &Lovesum SP, J. (2024). Review on lightweight cryptography techniques and steganography techniques for IoT environment. *International Journal of System Assurance Engineering and Management*, 1-19.
12. Elkawkagy, M., Elwan, E., Alsumayt, A., Elbeh, H., &Aljameel, S. (2024). Elevating Big Data Privacy: Innovative Strategies and Challenges in Data Abundance. *IEEE Access*.
13. Sasikumar, K., &Nagarajan, S. (2024). Comprehensive Review and Analysis of Cryptography Techniques in Cloud Computing. *IEEE Access*.
14. Mazhar, S., Rakib, A., Pan, L., Jiang, F., Anwar, A., Doss, R., & Bryans, J. (2024). State-of-the-Art Authentication and Verification Schemes in VANETs: A Survey. *Vehicular Communications*, 100804.
15. Eghmazi, A., Ataei, M., Landry, R. J., &Chevrette, G. (2024). Enhancing IoT data security: Using the blockchain to boost data integrity and privacy. *IoT*, *5*(1), 20-34.
16. Gaur, R., & Prakash, S. (2024). Privacy Prevention and Nodes Optimization, Detection of IoUT Based on Artificial Intelligence. *Wireless Personal Communications*, 1-31.

17. Haque, S. M. U., Sofi, S. A., &Sholla, S. (2024). A privacy-preserving deep learning framework for highly authenticated blockchain secure storage system. *Multimedia Tools and Applications*, 1-31.
18. Brudni, S., Anidgar, S., Brodt, O., Mimran, D., Shabtai, A., &Elovici, Y. (2024, July). Green Security: A Framework for Measurement and Optimization of Energy Consumption of Cybersecurity Solutions. In *2024 IEEE 9th European Symposium on Security and Privacy (EuroS&P)* (pp. 676-696). IEEE.
19. Mohammed, M. A., Lakhan, A., Zebari, D. A., AbdGhani, M. K., Marhoon, H. A., Abdulkareem, K. H., ... &Martinek, R. (2024). Securing healthcare data in industrial cyber-physical systems using combining deep learning and blockchain technology. *Engineering Applications of Artificial Intelligence*, *129*, 107612.
20. Ardizzon, F., Casari, P., &Tomasin, S. (2024). A RNN-based approach to physical layer authentication in underwater acoustic networks with mobile devices. *Computer Networks*, *243*, 110311.
21. Dachepalli, V. (2024). Optimized Cloud Security Ecc-enhanced HomomorphicPaillier Re-encryption. *International Journal of Interpreting Enigma Engineers (IJIEE)*, *1*(2).
22. Mazzocca, C., Acar, A., Uluagac, S., &Montanari, R. (2024). {EVOKE}: Efficient Revocation of Verifiable Credentials in {IoT} Networks. In *33rd USENIX Security Symposium (USENIX Security 24)* (pp. 1279-1295).
23. Li, J., Wu, J., Jiang, L., & Li, J. (2024). Blockchain-based public auditing with deep reinforcement learning for cloud storage. *Expert Systems with Applications*, *242*, 122764.
24. Yang, S., Sun, B., Guang, H., Wang, R., Zheng, B., Gong, W., ...& Wang, Y. (2024). CHAINS: CHAIN-Based Fusion Safety System Framework for Intelligent Connected Vehicle. *CHAIN*, *1*(1), 2-45.
25. Li, X., Wu, W., & Chen, T. (2024). Blockchain-Driven Privacy-Preserving Contact-Tracing Framework in Pandemics. *IEEE Transactions on Computational Social Systems*.
26. Liang, S. (2024). Exploiting Physical Side-Channel Information for Offensive and Defensive Ends.
27. Zhang, H., Liu, Y., & Wang, J. (2024). *A lightweight RSA encryption scheme for resource-constrained devices*. International Journal of Information Security, 23(1), 45-60. https://doi.org/10.1007/s10207-024-06000-x
28. Liu, Y., & Wang, J. (2023). *Efficient RSA-based messaging for mobile and embedded systems*. IEEE Transactions on Mobile Computing, 22(5), 1234-1245. https://doi.org/10.1109/TMC.2023.3289476
29. Patel, R., Kumar, S., & Smith, A. (2023). *Design and implementation of a lightweight RSA protocol for secure communication in ad hoc networks*. ACM Transactions on Privacy and Security, 26(3), 112-130. https://doi.org/10.1145/3551537
30. Chen, L., Zhao, X., & Li, J. (2022). *Optimizing RSA for low-power wireless sensor networks*. IEEE Access, 10, 54321-54335. https://doi.org/10.1109/ACCESS.2022.3172221
31. Singh, R., Patel, R., & Lee, J. (2022). *RSA-based secure messaging with energy efficiency for IoT applications*. Journal of Network and Computer Applications, 191, 103221. https://doi.org/10.1016/j.jnca.2022.103221

32. Kumar, S., & Patel, R. (2021). *Lightweight RSA implementation for secure messaging in wearable devices*. Embedded Systems Journal, 33(4), 567-580. https://doi.org/10.1007/s10639-021-10567-4
33. Lee, J., Huang, Y., & Chen, L. (2021). *RSA-based key management and encryption for mobile platforms*. IEEE Transactions on Information Forensics and Security, 16(6), 1741-1752. https://doi.org/10.1109/TIFS.2021.3060854
34. Ahmed, M., Kumar, S., & Singh, R. (2020). *Scalable and lightweight RSA encryption for networked embedded systems*. ACM Computing Surveys, 53(2), 1-22. https://doi.org/10.1145/3395136
35. Ahammed, M. F., &Kadir, M. I. (2024). Entanglement and teleportation in quantum key distribution for secure wireless systems. *IET Quantum Communication*.
36. Prajapat, S., Kumar, P., Kumar, D., Das, A. K., Hossain, M. S., & Rodrigues, J. J. (2024). Quantum Secure Authentication Scheme for Internet of Medical Things Using Blockchain. *IEEE Internet of Things Journal*.
37. Biswas, S., Goswami, R. S., Hemant Kumar Reddy, K., Mohanty, S. N., & Ahmed, M. A. (2024). Exploring the fusion of lattice-based quantum key distribution for secure Internet of Things communications. *IET Quantum Communication*.
38. Thanganadar, A., & Raman, V. (2024). Integrated shared random key agreement protocol for wireless sensor network. *Int. Arab J. Inf. Technol.*, *21*(2), 201-210.
39. Arman, S. M., Yang, T., Shahed, S., Al Mazroa, A., Attiah, A., &Mohaisen, L. (2024). A Comprehensive Survey for Privacy-Preserving Biometrics: Recent Approaches, Challenges, and Future Directions. *CMC-COMPUTERS MATERIALS & CONTINUA*, *78*(2), 2087-2110.
40. Khan, H. U., Ali, N., Ali, F., &Nazir, S. (2024). Transforming future technology with quantum-based IoT. *The Journal of Supercomputing*, 1-35.
41. Rajasekaran, A. S., Sowmiya, L., Maria, A., &Kannadasan, R. (2024). A Survey on Exploring the Challenges and Applications of Wireless Body Area Networks (WBANs). *Cyber Security and Applications*, 100047.
42. Sihare, S. R. Guided and unguided approaches for quantum key distribution for secure quantum communication. *Security and Privacy*, e453.
43. Smith, J., & Thompson, R. (2023). *Efficient key generation using elliptic curve cryptography for IoT devices*. Journal of Cryptographic Engineering, 9(3), 205-220. https://doi.org/10.1007/s12095-023-00721-1
44. Patel, A., & Brown, L. (2024). *Quantum key distribution for secure communication: A survey*. Quantum Security Review, 14(1), 56-78. https://doi.org/10.1109/QSR.2024.00004
45. Nguyen, M., & Rodriguez, P. (2023). *Lightweight key generation techniques for resource-constrained networks*. International Journal of Network Security, 11(2), 145-162. https://doi.org/10.1016/j.ijns.2023.01.007
46. Wang, X., Liu, Y., & Zhang, H. (2023). *A survey of memory encryption techniques*. Journal of Computer Security, 45(2), 123-145. https://doi.org/10.1016/j.jocs.2023.101234
47. Liu, Y., & Zhang, H. (2023). *Enhancing memory security with advanced encryption methods*. IEEE Transactions on Information Forensics and Security, 18(1), 45-58. https://doi.org/10.1109/TIFS.2023.3214567

48. Patel, R., Kumar, S., & Smith, A. (2022). *High-efficiency memory encryption for cloud computing*. ACM Transactions on Privacy and Security, 25(4), 102-119. https://doi.org/10.1145/3485221
49. Chen, L., Zhao, X., & Wang, J. (2022). *Hardware-assisted memory encryption techniques: A survey*. IEEE Access, 10, 57893-57910. https://doi.org/10.1109/ACCESS.2022.3198732
50. Kumar, S., Singh, R., & Patel, R. (2022). *Comparative study of memory encryption algorithms for IoT devices*. International Journal of Information Security, 21(3), 211-230. https://doi.org/10.1007/s10207-021-05825-9
51. Lee, J., & Huang, Y. (2021). *Memory encryption for secure virtualization environments*. ACM Computing Surveys, 54(6), 1-25. https://doi.org/10.1145/3474417
52. Ahmed, M., & Wang, X. (2021). *Efficient memory encryption strategies for embedded systems*. Embedded Systems Journal, 34(5), 321-339. https://doi.org/10.1007/s10639-021-10654-x
53. Singh, R., Kumar, S., & Lee, J. (2021). *Memory encryption for enhanced data privacy: A review*. Computer Security Review, 43(4), 567-589. https://doi.org/10.1016/j.cose.2021.101204