SSRG International Journal of Electronics and Communication Engineering
ISSN: 2348-8549/ https://doi.org/10.14445/23488549/IJECE-V13I2P107

Volume 13 Issue 2, 91-102, February 2026

Original Article

Lightweight Intrusion Detection System forSecure Data Transmission in Switched Network Environments

Ritu Rani[1], Rishi Pal Singh[2]

[1,2]*Department of Computer Science and Engineering, Guru Jambheshwar University of Science and Technology, Hisar, India.*

[1]*Corresponding Author : pal_rishi@yahoo.com*

Received: 17 November 2025 Revised: 19 December 2025 Accepted: 20 January 2026 Published: 20 February 2026

Abstract *- While increased internet connectivity offers numerous benefits, it also raises the risk of cyber-attacks by creating more points of vulnerability through system-to-system, system-to-server, and system-to-internet interactions. These interactions can be exploited to gain control over systems and their connected devices. Although security vulnerabilities and mitigation strategies have been extensively studied, there remains a significant gap in understanding how cyber-attacks leverage these vulnerabilities to compromise system performance and operations. In the context, we propose a Lightweight Intrusion Detection System (LIDS) for Secure Data Transmission in Switched Network Environments. The proposed system ensures the authenticity of data packets and prevents unauthorized nodes from joining the switched network. Specifically, the LIDS is capable of detecting both unauthorized and compromised nodes in the Switched Network Environments. The proposed LIDS is simulated extensively, and its performance is compared with state-of-the-art approaches. The results demonstrate that LIDS outperforms existing approaches based on the performance metrics evaluated in the simulation, and achieves a detection rate 5% greater than that of the existing approach.*

Keywords *- Security, Connected systems, Connectivity, Intrusion detection approach.*

1. Introduction

Attacks on wireless networks are exploited globally, affecting households and corporate buildings alike. Current wireless network attacks are highly persistent, passive, and intelligent, often bypassing ordinary security measures. Advanced security procedures, such as firmware upgrades, encryption, and updating legacy standards, require additional hardware and financial investments. Hardware-based strategies, like the received signal strength indicator, are expensive and imprecise. Software approaches, such as signature-based techniques, rely on extensive signatures, observable statistics in statistical methods, and significant computational overhead in probabilistic methods, failing if an access point has encryption. Designing client-side intrusion detection mechanisms requires considerable effort, resources, and extensive authorization lists. Therefore, an effective intrusion detection system is essential to identify such attacks. It is crucial that a system's security mechanisms prevent unauthorized access to its data and resources. However, detecting invasion attempts allows corrective actions to be taken to repair any damage, either immediately or later. Intrusion Detection Systems (IDS) identify and recognize unusual or unauthorized activities in the network. IDS may also analyze potentially suspicious network activities. In a network configuration, systems and routers connected over wireless networks can move autonomously and self-organize. The inherent characteristics of the wireless environment make it susceptible to various threats, including passive eavesdropping and aggressive interference. Unlike wired networks, which require physical access or bypassing multiple defense layers, wireless networks are vulnerable to attacks from any direction and can target all systems. Considering that compromised node attacks are highly damaging and challenging to detect, systems and network infrastructure must be prepared to operate in a non-trusting mode. Identifying malicious behavior on the network as soon as it enters can significantly reduce network damage.

Therefore, every network must include an intrusion detection system. The present research introduces a two-phase detection approach for identifying unauthorized activities in compromised services and systems operating within networks. The proposed method aims to identify unauthorized systems without relying on symmetric or asymmetric encryption techniques, digital signatures, sequence numbers, or timestamps. Malicious systems, or compromised systems, perform active attacks intended to cause harm to other systems, potentially resulting in a network outage. Compromised nodes can disrupt the proper functioning of a routing protocol by altering or generating fake routing information. The system responsible for transmitting erroneous routing information may be compromised or experiencing synchronization issues due to unpredictable physical movement. Zhou and Haas [6] proposed a distributed key management service independent of routing protocols. This method uses redundancies in the network topology to ensure reliable key management. The

basic idea is to enable key sharing utilization even when the ratio of compromised systems to total systems reaches its maximum threshold. Multiple studies on security detection techniques for wireless networks based on infrastructure have been conducted, such as [2, 8, 11, 14, 18, 44]. To prevent issues, distributed approaches like key generation and management have been used to ensure the reliability and accuracy of routing information [5, 7, 9, 19]. Intrusion prevention approaches, including encryption and authenticity, are typically the initial defense against actions aimed at compromising a network's integrity, reliability, and confidentiality. However, relying solely on intrusion prevention is insufficient as systems grow increasingly intricate, and security is frequently neglected. Intrusion detection acts as an additional layer of protection for network systems, enabling prompt countermeasures to mitigate damage upon detecting an intrusion.

Future authentication methods prioritize establishing connections between new users and networks. Authentication in computer networks refers to proving the identity of a device or user on a network. This measure ensures that only authorized devices and users can access network resources. Various authentication methods, such as passwords, biometric factors like fingerprints or facial recognition, and security tokens, can be used. In general, authentication plays an essential role in securing computer networks. Networks must implement measures to restrict access to authorized users and ensure these users access a safe network, as unauthorized access might cause permanent harm.

The primary purpose is to set up a session key for secure communication, mutual authorization, and non-repudiation [8, 5]. Most access control systems employ public key management systems to establish a secure connection between a person and a secret key with a digital certificate. These certificates contain the public key, identity, and additional information securely signed by an authorized third party. Certificate Authorities (CAs) are responsible for creating public key certificates used by applications. Security requirements for Certification Authorities are crucial, as they must consider and analyze the extensive array of potential attacks against them [1, 4].

Key management strategies for networks aim to eliminate the need for a centralized certificate authority. The first technique outlined in [6, 45, 46] replicates a traditional cryptographic algorithm by dispersing fragments of the confidential key across multiple systems. A key management strategy has been suggested for a network employing gateway cryptography and the public key framework. This approach includes distributing segments of the confidential key among specific systems designated as servers. To obtain the confidential key of a program or system, an attacker needs to breach a specific number of servers securely, known as the threshold. Periodic server share refreshes prevent gradual penetration of servers. Prior communication and coordination between the systems is necessary for establishing the service in this approach. Furthermore, it is impractical for every node to possess the public key of every other node in an ad hoc network when the number of nodes is large.

In the second technique [12], each system verifies the identity of the other using predetermined criteria, such as a shared secret among all nodes in a network. Every system in a network uses the shared secret to produce its unique keys. De Cleene et al. [3] developed a systematic structure as part of their scheme. Every level in the hierarchy is equipped with a controller. These area controllers reconfigure a system during transitions between distinguished regions. Kong et al. [12] have developed an alternative system that combines the emulation of a certifying authority and shared secret model with a centralized model based on a Public Key Infrastructure (PKI). Initially, the strategy includes an aerial node serving as the central node for distributing encryption keys. If the aerial node is destroyed, the system uses threshold cryptography, relying on secret sharing to act as a distributed certifying authority.

In the final technique, Hubaux et al. [11] introduced a trust-building technique for a network that relies on public-key distribution, similar to the PGP web of reliability. Unlike PGP, it lacks centralized certificate directories for certificate dissemination. Instead, a user selects a specific group of certificates from their collection to reveal to the other user. Both users then combine the certificates they have received with their unique certificates [45, 39]. To obtain a distant user's public key, the local user uses the Hunter Algorithm on the combined certificate repository to generate the certificate chain (s). The certificate reliability link should establish a connection between the client certificate and the remote user's certificate. The client uses the public key present in the remote user's certificate.

These operations include the security of links and networks, incorporating a detection structure consisting of two phases. In the first phase, the detection process verifies the actual identification of the communicating systems and identifies any unauthorized systems. In the second phase, the detection process examines whether the communicating systems are compromised. The main contribution of the paper is explained as follows:

1) A system model for a connected autonomous system environment is presented, focusing on the verification of new users and the detection of malicious users.
2) Unauthorized Node Detection (UND) and Compromised Node Detection (CND) algorithms are presented to identify malicious nodes and block them.
3) Mathematical models for both UND and CND are derived.
4) The proposed LIDS is simulated and tested to measure its performance against the contemporary methods, focusing on network performance and detection rate in a connected autonomous system environment.

2. Unauthorized Intrusion Detection System (IDS) Approach

In this section, we present a system model for autonomous systems, for new user verification and malicious system or user detection in the connected system model.

2.1. System Model

We consider a connected, autonomous system environment consisting of multiple systems or users, with an Intrusion Detection System (IDS) that monitors network traffic, detects unusual activity, and sends alerts when it occurs. It can detect any suspicious activities and alert the system administrator before any significant damage is done. An IDS is one of the most powerful methods for detecting computer network intrusions by monitoring unknown and suspicious activities.

2.2. System Verification for Authentication

In this approach, the sender vehicles broadcast their location information with its hash value to their neighboring system, and the receiving neighbors verify the integrity of the data or information. The sender vehicle shares its location information among neighboring vehicles for the purpose of routing. This section presents sender-side and receiver-side authentication.

2.2.1. Sender Side Authentication

The SA phase includes three sub-phases:
The Setup phase, the Key generation phase, and the Signature phase.

Setup Phase

The spatiotemporal data, such as the GPS coordinates of a system Gx, Gy, are generated, embedded in each system, and taken out and delivered to the next phase.

Key Generation Phase

Let a Trusted Authority (TA) be the public key generator that selects an additive group G with a large prime number q as the initial step. It determines a point P on the elliptic curve E. Z*p is a set of integers. It is assumed that the TA selects four hash functions as:

$$H_1 : (0, 1)* \rightarrow Z_p^* \text{ and } H_2 : (0, 1)* \rightarrow Z_p^* \quad (1)$$

$$H_3 : (0, 1)* \rightarrow Z_p^* \text{ and } H_4 : (0, 1)* \rightarrow Z_p^* \quad (2)$$

ret prime number $n_1 \in Z_p^*$ and makes a public key $P_u = n_1 P$.

Trusted Authority (TA) chooses a secret prime number n1∈ Zp *and makes a public key Pu=n1P. Then, TA distributes the following system parameters: (q, P, G, Pu, H1, H2, H3, H4). TA selects a prime number g ∈ Zq∗ and generates the private key Pr = gPu. TA sends this private key to new users through a secret communication channel.

Signature Phase

In the signature phase, first, each system in its transmission range floods Pu only to the neighboring user to which it belongs. Then, it calculates k = e (Gx + Gy) and r = gPuke. Four hash encryptions, which are based on encryption and hashing techniques, are applied to the message m to make the data packet more secure. It executes as follows:

$$m_1 = Enc_1(pu, m) \quad (3)$$

$$m_1 = Enc_1(pu, m) \quad (4)$$

$$m_3 = Enc_3(k, m_2) \quad (5)$$

$$m_4 = H_4(H_3(H_2(H_1(m_3||k||P)))) \quad (6)$$

Where $Enc1, Enc2, Enc3$ are encryption functions. The sender calculates $s = {}_{Pk}(e\ m4+e\ )k$ and floods (m, s) inside the cluster to which it belongs. The sending user disseminates $(m4, s)$ in its transmission range.

Hash functions are used to provide more security to the data packets (random-sized input and yield a fixed-size output) and also to verify the validity and integrity of data. In this paper, an elliptic curve is used to encrypt the message.

The sender takes the recipient's public key and performs a series of mathematical operations on the message using an elliptic curve to produce the ciphertext.

2.2.2. Receiver Side Authentication

In this section, the receiver user checks whether the intruder has manipulated the received data packet or not. The receiver user receives $(m4, s)$. In the verification phase, The receiving system calculates $rv\ ke^{m4}$ you Moreover, verify whether $r_v = r$ or not as follows:

$$rv = \frac{g^s}{ke^{m4}yu} \quad (7)$$

$$g(pkem4 + pkek) = kem4\ u \quad (8)$$

$$= \frac{(g^{p^{ke^{m4}}})(g^{p^{ke^k}})\ uu}{ke^{m4}p_u p^k e^k} \quad (9)$$

$$= g^u = r \quad (10)$$

If rv = r, the receiving user makes sure that the received message retains its integrity and the information in the message is not changed.

3. System Intrusion Detection Techniques (IDT)

When a malicious intruder joins the network and becomes successful in isolating any system, link failure occurs, which makes the whole path disconnected. There are also high chances that the malicious intruder compromises any sound system, and they also behave like a malicious intruder. In IDT, both unauthorized and compromised nodes are detected. For that reason, two security phases, such as the Unauthorized Node Detection (UND) phase and the Compromised Node Detection (CND) phase, are executed in IDT. In this section, UND and CND are presented for a connected network.

3.1. Unauthorized Node Detection (UND)

This phase detects the unauthorized node when systems or any new user is introduced in the network architecture for the first time. These nodes must be authenticated, and their identity must be checked properly before giving them all rights and privileges to access the network resources.

Because they are highly prone to malicious attacks and can sabotage the whole architecture, in Figure 1, let us say that A, B, and C are all authenticated systems. When a new system X1 wants to join the network, it has to be authenticated by the existing systems.

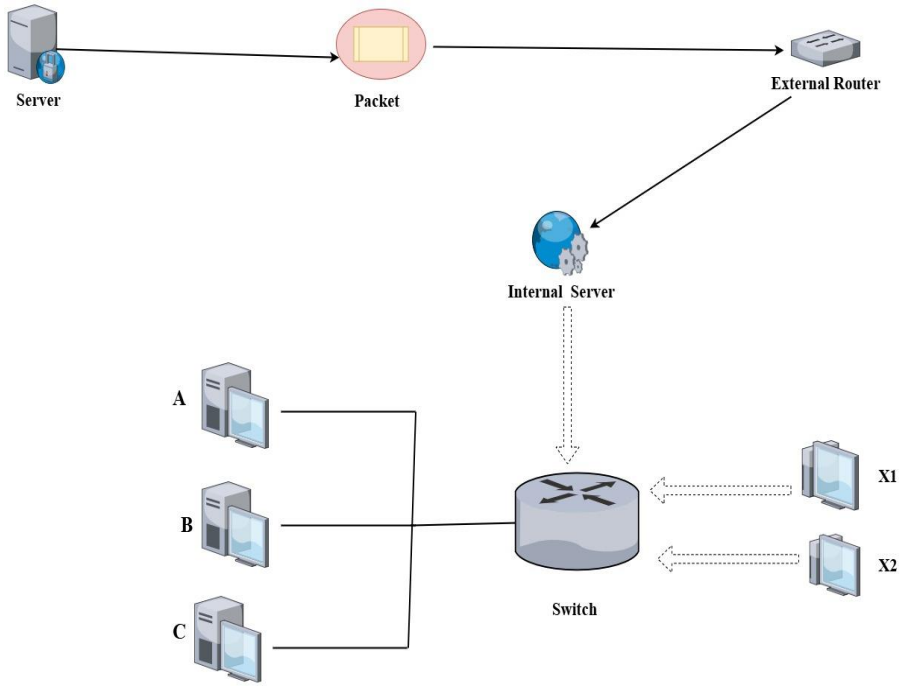


Fig. 1 Shows the network architecture

This phase detects the unauthorized node. When nodes are introduced in the network for the first time, they must be authenticated, and their identity must be checked properly before giving them all rights and privileges to access the network resources. Because they are highly prone to malicious attacks and can sabotage the whole architecture, in Figure 1 (a), let us say that A, B, and C are all authenticated systems.

When a new node X1 wants to join the network, it has to be authenticated by its neighbors. Suppose system A is out of X1's transmission range, and systems B and C are falling within each other's transmission range. So, only systems B and C have to authenticate it. When two nodes, X1 and X2, enter the network, they are both authenticated by their neighbors.

Firstly, node X1 joins the networks, and it is validated by both systems B and C. Secondly, when node X2 joins the secured network, nodes B and C are not in transmission range of X2. So, it is authenticated and validated by its neighbors X1 and A. The above procedure is corroborated as follows.

System $X1$ assumes the values of $y1, y2, y3, y4$ as $y1 = \beta 1x1, y2 = \beta 2x2, y3 = \beta 3x3,$ and $y4 = \beta 4x4$ respectively and calculates different possible values of yij as

$$y_{12} = y_1 + y_2 = x_1\beta_1 + x_2\beta_2 \quad (11)$$

$$y_{13} = y_1 + y_3 = x_1\beta_1 + x_3\beta_3 \quad (12)$$

$$y_{14} = y_1 + y_4 = x_1\beta_1 + x_4\beta_4 \quad (13)$$

$$y_{23} = y_2 + y_3 = x_2\beta_2 + x_3\beta_3 \quad (14)$$

$$y_{24} = y_2 + y_4 = x_2\beta_2 + x_4\beta_4 \quad (15)$$

$$y_{34} = y_3 + y_4 = x_3\beta_3 + x_4\beta_4 \quad (16)$$

$X1$ calculates different possible values of αi as:

$$\alpha1 = y12 + y13 = 2x1\beta1 + x2\beta2 + x3\beta \quad (17)$$

$$\alpha2 = y12 + y14 = 2x1\beta1 + x2\beta2 + x4\beta4 \quad (18)$$

$$\alpha3 = y12 + y23 = x1\beta1 + 2x2\beta2 + x3\beta3 \quad (19)$$

$$\alpha4 = y12 + y24 = x1\beta1 + 2x2\beta2 + x4\beta4 \quad (20)$$

$$\alpha5 = y12 + y34 = x1\beta1 + x2\beta2 + x3\beta3 + x4\beta4 \quad (21)$$

$$\alpha6 = y13 + y14 = 2x1\beta1 + x3\beta3 + x4\beta4 \quad (22)$$

$$\alpha7 = y13 + y23 = x1\beta1 + x2\beta2 + 2x3\beta3 \quad (23)$$

$$\alpha8 = y13 + y24 = x1\beta1 + x2\beta2 + x3\beta3 + x4\beta4 \quad (24)$$

$$\alpha9 = y13 + y34 = x1\beta1 + 2x3\beta3 + x4\beta4 \quad (25)$$

$$\alpha10 = y14 + y23 = x1\beta1 + x2\beta2 + x3\beta3 + x4\beta4 \quad (26)$$

$$\alpha11 = y14 + y24 = x1\beta1 + x2\beta2 + 2x4\beta4 \quad (27)$$

$$\alpha12 = y14 + y34 = x1\beta1 + x3\beta3 + 2x4\beta4 \quad (28)$$

$$\alpha13 = y23 + y24 = 2x2\beta2 + x3\beta3 + x4\beta4 \quad (29)$$

$$\alpha14 = y23 + y34 = x2\beta2 + 2x3\beta3 + x4\beta4 \quad (30)$$

$$\alpha15 = y24 + y34 = x2\beta2 + x3\beta3 + 2x4\beta4 \quad (31)$$

The following information is passed between A, B, and X1:

$$X1 \rightarrow B, C: m1 = e-(x1\beta1)\, e-(x3\beta3) \quad (32)$$

$$m1 = e(x3\beta3-x1\beta1) \quad (33)$$

$$B, C \rightarrow X1 : M2 = H(\beta1, \beta3, x1, x3, m1) \quad (34)$$

$$X1 \rightarrow B, C: m3 = \alpha14 - \alpha5 + km2(\alpha9 - \alpha15) \quad (35)$$

$$B, C : m4 = \frac{em4 \quad (ex2\beta2+x4\beta4)km2}{e(x3\beta3+x1\beta1)\, km2} \quad (36)$$

First, B, C checks whether m4/m5 equals m1 as follows:

$$m4 = \frac{e\alpha14-\alpha5+km2(\alpha9-\alpha15)\, e(x2\beta2+x4\beta4)\, km2}{\overline{e(x3\beta3+x1\beta1)\, km2}} \quad (37)$$

$$m4 = \frac{e\alpha14\, e(\alpha9)km2 \;\; e(x2\beta2+x4\beta4)km2}{e\alpha5 \;\; e(\alpha15)km2\, e(x3\beta3+x1\beta1)km2} \quad (38)$$

Now putting the values of α9, α14, α5, α15 we get

$$m4 = e(x3\beta3-x1\beta1) = m1 \quad (39)$$

Secondly, checks whether m5 equals m2 or not as follows:

$$m5 = H(\beta1, \beta3, x1, x3, m4) = H(\beta1, \beta3, x1, x3, m4) = m2 \quad (40)$$

Above satisfies the equation:

$$ky1 + ky2 + ky3 + ky4 = P \quad (41)$$

$$y1 + y2 + y3 + y4 = \frac{P}{k} \quad (42)$$

$$x1\beta1 + x2\beta2 + x3\beta3 + x4\beta4 = \frac{P}{k} \quad (43)$$

Hence, system X_1 is an authenticated user, and finally, it is accepted in the system's architecture. Similarly, user X_2 is validated by its neighbors and accepted into the network.

It helps to enhance the link connectivity. The whole idea of the UND is presented in Algorithm 1 to simplify the logic given above.

3.2. Pseudocode of UND ()

Input: y1, y2, y3, y4
Process
1 Unauthorized Node Detection [y1, y2, y3, y4]
2 Authenticate the new node by the already authenticated neighboring nodes existing in the direct communication range.
3 New node calculates message m1 using equation (33)
4 Neighboring nodes calculate message m2 using equation (34)
New node calculates message m3 using equation (35)
5 Neighboring nodes calculate message m4 and m5 using equations (36) and (38)
6 Neighboring nodes validate the new node using equations (39) and (40)
7 if (m4==m1 && m5==m2) then
8 authorized nodes
9 else Node is unauthorized.
10 end if
11 end unauthorized node.

3.3. Compromised Node Detection (CND)

This phase begins as soon as users start forwarding data. As the data disseminationuses a multi-hop strategy, the integrity status of nodes (i.e., whether compromised or not) It can be checked in every single hop. Because nodes are checked at every hop, the detection procedure is independent of velocity.

1. The compromised status of each system is determined by the server.
2. The server stores all relevant information, such as records of each node, itsthreat perception, and behavior, which does not match the previous norms.
3. Then, the following information is passed between A, B, and X, as follows:

$$X_1 \rightarrow B, C: \quad m_1 = e^{(x_3 e \beta_3 - x_1 e \beta_1)}\, e^{f(x1,x2)} \tag{44}$$

$$B, C \rightarrow X_1: \quad m_2 = H(\beta_1, \beta_3, m_1) \tag{45}$$

$$X_1 \rightarrow B, C: \quad m_3 = \alpha_{14} - \alpha_5 + km_2(\alpha_9 - \alpha_{15}) + f(x_1, x_2) \tag{46}$$

$$B, C: m_4 = \frac{e^{m_3}\, e^{(x_2\beta_2 + x_4\beta_4)^{km_2}}}{e^{(x_3\beta_3 + x_1\beta_1)_{km_2}}} \tag{47}$$

$$B, C: \quad m_5 = H(\beta_1, \beta_3, x_1, x_3, m_4) \tag{48}$$

First B, C checks whether m_4 equals m_1 as follows:

$$m_4 = \frac{e^{\alpha 14 - \alpha 5 + km2(\alpha 9 - \alpha 15 + f\ (x1,x2))} e^{(x2\beta 2 + x4\beta 4)^{km2}}}{e^{(x3\beta 3 + x1\beta 1)^{\ km2}}} \tag{49}$$

$$m_4 = \frac{e^{\alpha_{14}}\, e^{(\alpha_9)^{km_2}}\, e^{f(x_1,x_2)} e^{(x_2\beta_2 + x_4\beta_4)^{km_2}}}{e^{\alpha_5} e^{(\alpha_{15})^{\ km_2}}\, e^{(x_3\beta_3 + x_1\beta_1)_{\ km_2}}} \tag{50}$$

Put values of α_9, α_5, α_{14}, α_{15}

$$m_4 = e^{(x_3\beta_3 - x_1\beta_1)}\, e^{f(x_1, x_2)} = m_1 \tag{51}$$

Secondly, B, C checks whether m_5 equals m_2 as follows:

$$m_5 = H(\beta_1, \beta_3, x_1, x_3, m_4) = H(\beta_1, \beta_3, x_1, x_3, m_1) = m_2 \tag{52}$$

Here, f (x1, x2) is a chi-square distribution function used by the server to compute the confidence interval for determining if a node is compromised. If f (x1, x2) falls within the confidence interval, the system is considered not compromised. An algorithm is presented to simplify the IDT steps.

3.4. Pseudocode of CND ()

Input: (y1, y2, y3, y4)
Process
1 Compromised Node Detection (y1, y2, y3, y4)
2 New node calculates message m1 using equation (44) and transmits to neighboring systems.
3 Neighboring systems calculate message m2 using equation (45) and transmit it to the new node.
4 New nodes calculate message m3 using equation (46) and transmit to neighboring systems.
5 Neighboring systems calculate message m4 and m5 using equations (47) and (48).
6 Neighboring systems validate the new node using equations (51) and (52)
7 if (m4==m1&& m5==m2) then
8 Uncompromised Node
9 else node is compromised.
10 end if
11 end compromised node detection.

3.5. Computational Complexity Analysis

Computational number theory is a branch of number theory that studies algorithms and their complexity for solving number-theoretic problems. It has numerous applications in cryptography, coding theory, and other areas of computer science. In this section, we will provide an overview of computational number theory, its importance in modern applications, and the challenges researchers face in this field. Computational number theory involves the study of algorithms for solving problems related to integers and modular forms. Let us assume there are N nodes in the network. It is assumed that each node has M neighbours. In the validation of a new node, there are two comparisons for each new node. Therefore, the Time complexity will be O (N.2M) & O(NM). Time complexity is typically expressed using big O notation, which provides an upper bound on the growth rate of the algorithm's running time.

4. Simulation and Results Analysis

In this section, the performance evaluation of the proposed LIDS is conducted through simulations in switched network environments, where intruder/malicious nodes are present. A virtual simulation environment for switched networks is developed by creating unique ns-2 scripts for several modules, including UND, CND, key generation, and packet transmission. The simulation parameters used to conduct the experiments are given in Table 1. The proposed LIDS system is compared with the CMGV system.

4.1. Simulation Environment

The robustness of the LIDS is measured with two types of intruder nodes. The first type of intruder nodes is prohibited from entering the switched network.

The second type of intruder nodes are intelligent enough to enter the network, but they are detectable. We considered the Giga bits wireless channel to provide communication among nodes. All simulation parameters are shown in Table 1.

Table 1. Simulation parameters

Parameter	Value
No. of nodes	100
No. of intruders	20
No. of Packets Transmitted	100000
Size of Packet	512 Bytes
Transmission Range	100m
Packet Rate	0.1 pkt/s
Channel Capacity	1 Gbps
Traffic Model	CBR
Routing Protocol	AODV

4.2. Simulation Matrices

- Packet Delivery Ratio (PDR): The Packet Delivery Ratio is calculated by dividing the count of packets that are received successfully by the total count of packets that are sent. It serves as a measurement for the proportion of packets sent from its origin that reach their destination without any errors or losses.
- Throughput: Throughput is the mean amount of data that can effectively pass through the network within a specific period of time. It also defines the speed at which messages are transmitted over a communication channel, such as Ethernet, in a communication network.
- End-to-End Delay: The end-to-end delay refers to the time it takes for a data packet to transit from its origin to its destination within a network.
- Detection Rate: The detection rate is the percentage of anomalies found out of the total number of anomalies in the data. As the number of intruders increases, the detection rate of the proposed LIDS protocol decreases.
- Packet Loss: Packet loss is the failure of a network packet to reach its final destination, resulting in the loss of information—packets, which are discrete units.

Data are sent and received during the process of connecting to any network. Packet loss describes the situation where one or more packets do not successfully reach the place they were intended to go.

4.3. Simulation Results

The simulation results are obtained for the proposed LIDS that uses the CAG [4] as the routing approach. The number of malicious systems considered is 2 and 10 for simulation purposes. The simulation results are compared with CMGV, AOPRF [48], and LNID [47].

4.3.1. Packet Delivery Ratio (PDR) Analysis

Figure 2 shows the packet delivery ratio in the network with 10 intruders. The proposed LIDS protocol demonstrates an increasing packet delivery ratio as the number of nodes increases. This improvement occurs because LIDS can detect and remove intruders more efficiently, leading to smoother communication and fewer interruptions. In contrast, the CMGV protocol also shows an increase in packet delivery ratio with more nodes, but this increase is less significant than that of LIDS. The better performance of LIDS is due to its advanced intruder detection and removal capabilities. This allows for higher successful data transmissions even in the Presence of multiple intruders. As a result, LIDS ensures a more reliable network performance, maintaining better packet delivery ratios as the network grows. The efficiency of LIDS in handling intruders highlights its robustness and effectiveness in maintaining communication quality under challenging conditions. Therefore, LIDS provides a more dependable solution for networks with a high number of intruders, ensuring consistent and reliable data delivery. When we consider a number of nodes 100 and intruder =2, the PDR for CMGV is 65%. When the number of intruders increase to 10, the PDR decreases to 40%; while the PDR of LIDS with 2 intruders is 98% and it decreases to 50% with 10 intruders, but it is better than CMGV.

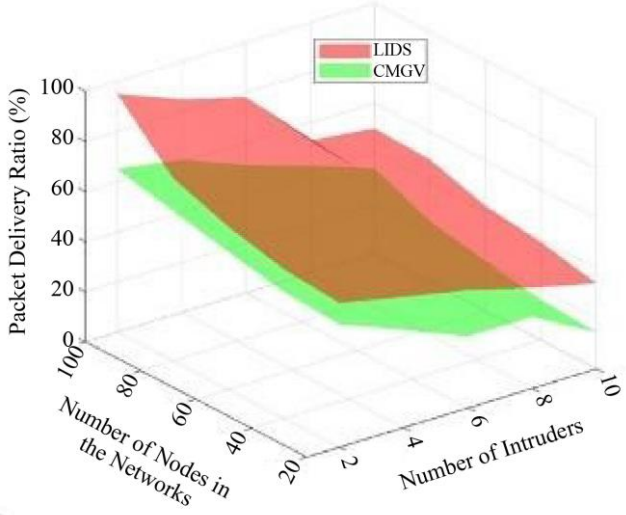


Fig. 2 PDR with different number of nodes in the presence of intruders

4.3.2. End-to-End Delay Analysis

Figure 3 shows the end-to-end delay in the network when 10 intruders are present. The proposed LIDS protocol effectively reduces this delay as the number of intruders increases. This reduction is achieved through LIDS's efficient handling of network congestion and improved routing strategies, which ensure data packets are delivered faster. In comparison, the CMGV protocol also decreases the end-to-end delay with more intruders, but not as significantly as LIDS. The superior performance of LIDS means it can maintain lower latency, or delay, even when the network is heavily intruded upon. This results in more reliable communication. Overall, LIDS demonstrates better adaptability and efficiency under challenging network conditions with multiple intruders. End-to-end delay of the LIDS model is decreasing as the number of intruders increases in the networks compared to CMGV. For example, when the number of intruders is 6, the end-to-end delay of LIDS is 150 ms; when the number of intruders increases to 10, the end-to-end delay is around 180. While for CMGV with 6 intruders is 300 ms, when the number of intruders increases to 10, the end-to-end delay is above 400. It is clearly recognized that the end-to-end delay is less than the model used for comparison.

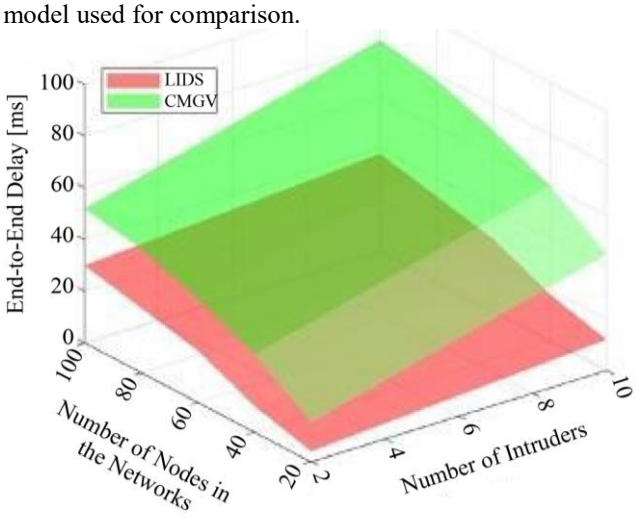


Fig. 3 End-to-end with different numbers of nodes in the presence of intruders

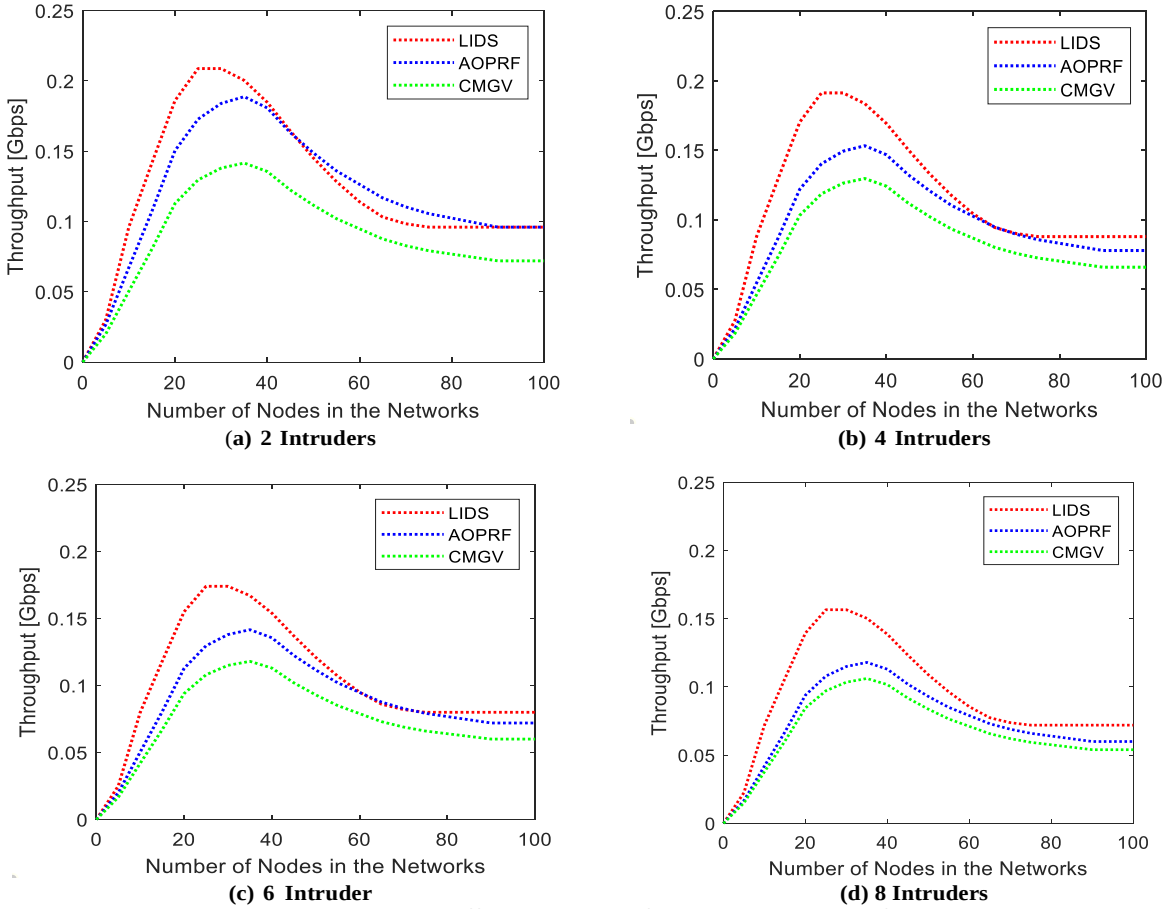
4.3.3. Throughput

The throughput of a network, which indicates how efficiently data is transmitted, varies with the number of nodes and the Presence of intruders. More nodes generally allow the network to handle higher data traffic. However, intruders can disrupt this performance. With fewer nodes, the throughput is lower due to limited capacity. As the number of nodes increases, the network can initially improve its throughput by managing the data flow more effectively. However, the Presence of intruders can interfere with this improvement. Intruders cause interference and data loss, reducing the overall throughput. Advanced protocols like LIDS can mitigate some of these disruptive effects, maintaining better throughput compared to other protocols like AOPRF and CMGV. This demonstrates the importance of robust network protocols to manage data transmission efficiently, even in the Presence of intruders.

As the number of intruders in a network rises, the throughput usually goes down. This happens because intruders cause extra interference and congestion, disrupting the normal flow of data. Their actions can lead to more packet loss, delays, and collisions, which further hurt network performance. The problem gets worse in networks

with heavy traffic and many nodes, as these systems struggle to handle and recover from the disruptions. As a result, keeping high throughput becomes more difficult. Advanced network protocols are needed to manage these issues and keep performance up.

As shown in Figures 4 (a-d), when the number of intruders increases from 2 to 8, network congestion intensifies, resulting in higher data collision and retransmission rates. Intruders deplete bandwidth and processing power, diminishing resources available for legitimate traffic. The additional overhead from managing intruders further strains the network. Consequently, the overall throughput decreases significantly in the proposed model LIDS. However, the throughput of AOPRF and CMGV also decreases as the number of intruders increases from 2 to 8. The rate of decrement in throughput is less for the proposed LIDS as compared to both AOPRF and CMGV. This is due to the fact that in LIDS, the UND algorithm detects the intruder and prevents them from entering the networks.



(a) 2 Intruders **(b) 4 Intruders**

(c) 6 Intruder **(d) 8 Intruders**

Fig. 4 Throughput with different numbers of nodes in the presence of intruders

4.3.4. Detection Rate

Figure 5 shows the detection rate in a network with 100 nodes. This happens because it becomes more challenging to identify each intruder accurately. The detection rate obtained through the proposed LIDS is compared with the contemporary methods: AOPRF, LNID, and CMGV. It is seen that the detection rate of AOPRF is gradually increasing as the number of nodes increases, but the detection rate is stable after 80 nodes, and it is around 94%. However, the proposed LIDS has a detection rate of around 95%. The remaining two methods, LNID and CMGV, achieve a detection rate lower than LIDS. The CMGV protocol also sees a drop in detection rate with more intruders, but this decrease is not as steep as with LIDS. Even though LIDS's detection rate drops, it still performs better overall than CMGV in spotting intruders, even when there are many of them. This indicates that LISD is more effective in maintaining a higher detection rate under challenging conditions. Therefore, LIDS provides more reliable intruder detection in large networks, ensuring better security and monitoring compared to CMGV. LIDS model detection rate is the same as CMGV model with 40 nodes but LIDS achieving detection rate 5% greater than that of the existing approach with 100 nodes. It is clearly observed that the proposed LIDS detects a higher number of intruders as compared with AOPRF, LNID, and CMGV. It is because the UND algorithm identifies false position information and prevents the intruder from becoming part of the network. If the intruder somehow enters the network, the CND algorithm detects it and blocks the intruder.

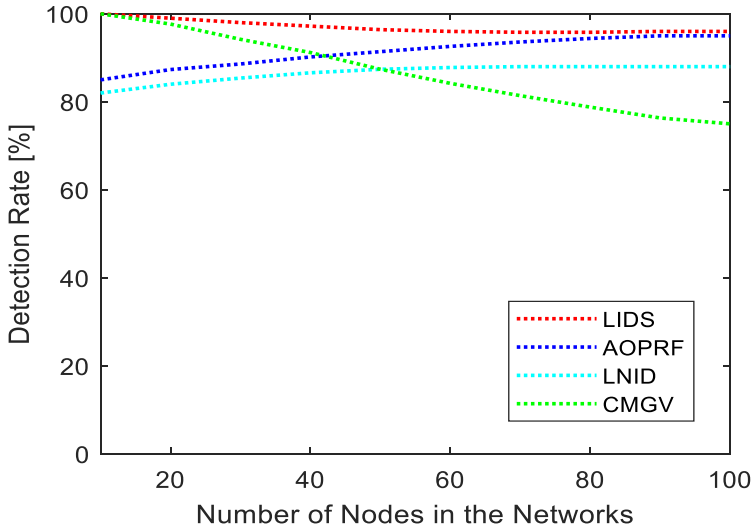


Fig. 5 Detection rate with number of nodes in the presence of intruders

5. Conclusion and Future Work

This paper proposes a Lightweight Intrusion Detection System (LIDS) for Secure Data Transmission in Switched Network Environments. The proposed LIDS is based on a two-phase detection approach for identifying and recognizing unauthorized activities in compromised services and systems operating within networks. In the proposed approach, key generation techniques, measurements, and core principles are employed to generate the key. The key generation and packet transmission are carried out using the CAG [4] as the routing approach. A promising approach for securely distributing secret keys among authorized users leverages the inherent randomness of wireless communication. This paper examines the challenges related to wireless network security, specifically focusing on key generation techniques, measurements, and core principles.

Additionally, we discuss methods to optimize key generation performance. We analyzed various application scenarios to better understand different environments' unique characteristics and obstacles. Although there have been significant advancements in this field, some issues that impact the reliability of key generation remain unresolved. The simulation results demonstrate that LIDS achieves a detection rate 5% greater than that of the existing approach.

The simulation results show that LIDS demonstrates better adaptability and efficiency in maintaining communication quality under challenging network conditions. Future research directions include developing group key generation methods, enhancing security against attacks on key generation systems, and creating key generation techniques for static environments. In the future, we will also use a generator for a high-speed network to identify unauthorized nodes and compromised nodes. This demonstrates that advanced network protocols like LIDS can mitigate some of these disruptive effects, maintaining better throughput compared to other protocols like CMGV. In the future, to propose IDS with potential datasets, we may employ Deep Learning or Machine Learning approaches to give more efficient results.

References

[1] Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Vanstone, *Hand- Book of AppliedCryptography*, CRC Press, 2001. [Google Scholar] [Publisher Link]

[2] A. Mishra, K. Nadkarni, and A. Patcha, "Intrusion Detection in Wireless ad Hoc Networks,"*IEEE Personal Communications*, vol. 11, no. 1, pp. 48-60, 2004. [CrossRef] [Google Scholar] [Publisher Link]

[3] B. DeCleene et al., "Secure Group Communications for Wireless Networks," *2001 MILCOM Proceedings Communications for Network-Centric Operations: Creating the Information Force (Cat. No.01CH37277)*, McLean, VA, USA, vol. 1, pp. 113-117, 2001. [CrossRef] [Google Scholar] [Publisher Link]

[4] Christos Douligeris, and Aikaterini Mitrokotsa, "DDoS Attacks and Defense Mechanisms: Classification and State-of-the-art," *Computer Networks*, vol. 44, no. 5, pp. 643-666, 2004. [CrossRef] [Google Scholar] [Publisher Link]

[5] Sushil Kumar et al., "Cybersecurity Measures for Geocasting in Vehicular Cyber Physical System Environments," *IEEE Internet of Things Journal*, vol. 6, no. 4, pp. 5916-5926, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[6] Daniel Krajzewicz et al., "Recent Development and Applications of SUMO – Simulation of Urban Mobility," *International Journal on Advances in Systems and Measurements*, vol. 5, no. 3&4, pp. 128-138, 2012. [Google Scholar] [Publisher Link]

[7] D. Watkins, and C. Scott, "Methodology for Evaluating the Effectiveness of Intrusion Detection in Tactical Mobile Ad-hoc Networks," *2004 IEEE Wireless Communications and Networking Conference (IEEE Cat. No.04TH8733)*, Atlanta, GA, USA, vol. 1, pp. 622-627, 2004. [CrossRef] [Google Scholar] [Publisher Link]

[8] Nnanna Ekedebe et al., "On a Simulation Study of Cyber Attacks on Vehicle-to-Infrastructure Communication (V2I) in Intelligent Transportation System (ITS)," *Proceedings Mobile Multimedia/Image Processing, Security, and Applications*, vol. 9497, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[9] E.C.H. Ngai, and M.R. Lyu, "Trust- and Clustering-based Authentication Services in Mobile Ad Hoc Networks," *24th International Conference on Distributed Computing Systems Workshops*, Tokyo, Japan, pp. 582-587, 2004. [CrossRef] [Google Scholar] [Publisher Link]

[10] George P. Corser, Huirong Fu, and Abdelnasser Banihani, "Evaluating Location Privacy in Vehicular Communications and Applications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 9, pp. 2658-2667, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[11] Yue Cao et al., "A Reliable and Efficient Encounter-Based Routing Framework for Delay/Disruption Tolerant Networks," *IEEE Sensors Journal*, vol. 15, no. 7, pp. 4004-4018, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[12] Nikos Komninos, Dimitris Vergados, and Christos Douligeris, "Detecting Unauthorized and Compromised Nodes in Mobile Ad Hoc Networks," *Ad Hoc Networks*, vol. 5, no. 3, pp. 289-298, 2007. [CrossRef] [Google Scholar] [Publisher Link]

[13] N. Komninos, "*Security Architecture for Future Communication Systems*," PhD Thesis, Lancaster University, 2003. [Google Scholar]

[14] Jiejun Kong et al., "Adaptive Security for Multi-layer Ad-hoc Networks," *Wireless Communications and Mobile Computing*, vol. 2, pp. 533-547, 2002. [CrossRef] [Google Scholar] [Publisher Link]

[15] Amith Murthy, Muhammad Rizwan Asghar, and Wanqing Tu, "A Lightweight Intrusion Detection for Internet of Things-based Smart Buildings," *Security and Privacy*, vol. 7, no. 4, pp. 1-18, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[16] Q. Xue, J. Sun, and Z. Wei, "TJIDS: An Intrusion Detection Architecture for Distributed Network," *CCECE 2003 - Canadian Conference on Electrical and Computer Engineering. Toward a Caring and Humane Technology (Cat. No.03CH37436)*, Montreal, QC, Canada, pp. 709-712, 2003. [CrossRef] [Google Scholar] [Publisher Link]

[17] W. Zhang et al., "Secure Routing in Ad Hoc Networks and a Related Intrusion Detection Problem," *IEEE Military Communications Conference, 2003. MILCOM 2003*, Boston, MA, USA, vol. 2, pp. 735-740, 2003. [CrossRef] [Google Scholar] [Publisher Link]

[18] Muhammad Mahmudul Islam, Ronald Pose, and Carlo Kopp, "An Intrusion Detection System for Suburban Ad-hoc Networks," *TENCON 2005 - 2005 IEEE Region 10 Conference*, Melbourne, VIC, Australia, pp. 1-6, 2005. [CrossRef] [Google Scholar] [Publisher Link]

[19] Yongguang Zhang, and Wenke Lee, "Intrusion Detection in Wireless Ad-hoc Networks," *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking*, Boston Massachusetts USA, pp. 275-283, 2000. [CrossRef] [Google Scholar] [Publisher Link]

[20] G. Vigna et al., "An Intrusion Detection Tool for AODV-based Ad hoc Wireless Networks," *20th Annual Computer Security Applications Conference*, Tucson, AZ, USA, pp. 16-27, 2004. [CrossRef] [Google Scholar] [Publisher Link]

[21] Gang Xu, and L. Iftode, "Locality Driven Key Management Architecture for Mobile Ad-hoc Networks," *2004 IEEE International Conference on Mobile Ad-hoc and Sensor Systems (IEEE Cat. No.04EX975)*, Fort Lauderdale, FL, USA, pp. 436-446, 2004. [CrossRef] [Google Scholar] [Publisher Link]

[22] K. Ilgun, R.A. Kemmerer, and P.A. Porras, "State Transition Analysis: A Rule-based Intrusion Detection Approach," *IEEE Transactions on Software Engineering*, vol. 21, no. 3, pp. 181-199, 1995. [CrossRef] [Google Scholar] [Publisher Link]

[23] Yuvraj Singh, and Sanjay Kumar Jena, "Intrusion Detection System for Detecting Malicious Nodes in Mobile Ad Hoc Networks," *First International Conference on Advances in ParallelDistributed Computing Technologies and Applications*, Tirunelveli, Tamil Nadu, India, pp. 410-419, 2011. [CrossRef] [Google Scholar] [Publisher Link]

[24] Satria Mandala, Md. Asri Ngadi, and A. Hanan Abdullah, "A Survey on MANET Intrusion Detection" *International Journal of Computer Science and Security*, vol. 2, no. 1, pp. 1-11, 2008. [Google Scholar] [Publisher Link]

[25] Ningrinla Marchang, and Raja Datta, "Collaborative Techniques for Intrusion Detection in Mobile Ad-hoc Networks," *Ad Hoc Networks*, vol. 6, no. 4, pp. 508-523, 2008. [CrossRef] [Google Scholar] [Publisher Link]

[26] A. Rajaram, and Dr. S. Palaniswami, "Malicious Node Detection System for Mobile Ad hoc Networks," *International Journal of Computer Science and Information Technologies*, vol. 1, no. 2, pp. 77-85, 2010. [Google Scholar] [Publisher Link]

[27] Nur Al Hasan Haldar, Muhammad Abulaish, and Syed Asim Pasha, "An Activity Pattern Based Wireless Intrusion Detection System," *2012 Ninth International Conference on Information Technology-New Generations*, Las Vegas, NV, USA, pp. 846-847, 2012. [CrossRef] [Google Scholar] [Publisher Link]

[28] Robert Koch, "Towards Next-Generation Intrusion Detection," *2011 3rd International Conference on Cyber Conflict*, Tallinn, Estonia, pp. 1-18, 2011. [Google Scholar] [Publisher Link]

[29] Kaushik Das, IPSec & IPv6 - Securing the NextGen Internet. [Google Scholar] [Publisher Link]

[30] Anup K. Ghosh, Christoph Michael, and Michael Schatz, "A Real-Time Intrusion Detection System Based on Learning Program Behavior," *Third International Workshop on Recent Advances in Intrusion Detection*, Toulouse, France, pp. 93-109, 2000. [CrossRef] [Google Scholar] [Publisher Link]

[31] Kamaldeep et al., "IoT-Sentry: A Cross-Layer-Based Intrusion Detection System in Standardized Internet of Things," *IEEE Sensors Journal*, vol. 21, no. 24, pp. 28066-28076, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[32] Cherifa Hamroun et al., "Intrusion Detection in 5G and Wi-Fi Networks: A Survey of Current Methods, Challenges, and Perspectives," *IEEE Access*, vol. 13, pp. 40950-40976, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[33] Lorenzo Diana, Pierpaolo Dini, and Davide Paolini, "Overview on Intrusion Detection Systems for Computers Networking Security," *Computers*, vol. 14, no. 3, pp. 1-44, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[34] Thomas D. Garvey, and Teresa F. Lunt, "Model Based Intrusion Detection," *Proceedings of the 14th National Computer Security Conference*, Omni Shoreham Hotel Washington, D.C., pp. 372-385, 1991. [Google Scholar] [Publisher Link]

[35] Vaishnavi Sivagaminathan, Manmohan Sharma, and Santosh Kumar Henge, "Intrusion Detection Systems for Wireless Sensor Networks using Computational Intelligence Techniques," *Cybersecurity*, vol. 6, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[36] Asma Alotaibi, and Ahmed Barnawi, "IDSoft: A Federated and Softwarized Intrusion Detection Framework for Massive Internet of Things in 6G Network," *Journal of King Saud University-Computer and Information Sciences*, vol. 35, no. 6, pp. 1-13, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[37] Jeng-Shyang Pan et al., "A Lightweight Intelligent Intrusion Detection Model for Wireless Sensor Networks," *Security and Communication Networks*, vol. 2021, pp. 1-15, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[38] Manohar Srinivasan, and N. C. Senthilkumar, "Intrusion Detection and Prevention System (IDPS) Model for IIoT Environments Using Hybridized Framework," *IEEE* Access, vol. 13, pp. 26608-26621, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[39] D. Sivakumar, and B. Sivakumar, "Detection and Localization of Attackers in Wireless Networks," *International Review on Computers and Software (IRECOS)*, vol. 9, no. 5, pp. 854-864, 2014. [Google Scholar] [Publisher Link]

[40] Ibrahim Al Shourbaji, and Rafat AlAmeer, "Wireless Intrusion Detection Systems (WIDS)," *Advances in ComputerScience and its Applications (ACSA)*, vol. 2, no. 3, pp. 1-6, 2013. [CrossRef] [Google Scholar] [Publisher Link]

[41] M. Ali Aydın, A. Halim Zaim, and K. Gökhan Ceylan, "A Hybrid Intrusion Detection System Design for Computer Network Security," *Computers & Electrical Engineering*, vol. 35, no. 3, pp. 517-526, 2009. [CrossRef] [Google Scholar] [Publisher Link]

[42] Salman Muhammad, Bagio Budiardjo, and Kalamullah Ramli, "Key Issues and Challenges of Intrusion Detection and Prevention System: Developing Proactive Protection in Wireless Network Environment," *World Academy of Science, Engineering and Technology*, vol. 77, pp. 521-524, 2011. [Google Scholar]

[43] Ioannis Krontiris, Thanassis Giannetsos, and Tassos Dimitriou, "LIDeA: A Distributed Lightweight Intrusion Detection Architecture for Sensor Networks," *Proceedings of the 4th International Conference on Security and Privacy in Communication Networks*, Istanbul Turkey, pp. 1-10, 2008. [CrossRef] [Google Scholar] [Publisher Link]

[44] Sana Ullah Jan et al., "Toward a Lightweight Intrusion Detection System for the Internet of Things," *IEEE Access*, vol. 7, pp. 42450-42471, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[45] Ismail Butun, Salvatore D. Morgera, and Ravi Sankar, "A Survey of Intrusion Detection Systems in Wireless Sensor Networks," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 266-282, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[46] Mustafa Al-Tahee et al., "Artificial Intelligence Assisted Cyber-Physical Systems with Intelligent Cyber Security Using Deep Learning," *2024 International Conference on Smart Systems for Electrical, Electronics, Communication and Computer Engineering (ICSSEECC)*, Coimbatore, India, pp. 689-694, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[47] Li He, "Design of a Lightweight Network Intrusion Detection System Based on Artificial Intelligence Technology," *Journal of Cyber Security and Mobility*, vol. 13, no. 5, pp. 1129-1148, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[48] ANKIT Manderna et al., "Intrusion Detection in Internet of Things using Differential Privacy: A Hybrid Machine Learning Approach," *Ad Hoc Networks*, vol. 174, 2025. [CrossRef] [Google Scholar] [Publisher Link]