MASTER UNIVERSITARIO EN INGENIERÍA DE TELECOMUNICACIONES

TRABAJO FIN DE MASTER

Location Deception in Ad Hoc Wireless Networks

Autor: Jaime Mohedano Aragon

Director: Dr. Gang Qu

Madrid

Febrero 2025

Declaro, bajo mi responsabilidad, que el Proyecto presentado con el título

Location Deception in Ad Hoc Wireless Networks

en la ETS de Ingeniería - ICAI de la Universidad Pontificia Comillas en el curso académico 2024/25 es de mi autoría, original e inédito y no ha sido presentado con anterioridad a otros efectos.

El Proyecto no es plagio de otro, ni total ni parcialmente y la información que ha sido tomada de otros documentos está debidamente referenciada.

Fdo.: Jaime Mohedano Aragón Fecha: 03./ 02./ 2025

Autorizada la entrega del proyecto

EL DIRECTOR DEL PROYECTO

Fdo.: Dr. Gang Qu Fecha: 04./ 02./ 2025

MASTER UNIVERSITARIO EN INGENIERÍA DE TELECOMUNICACIONES

TRABAJO FIN DE MASTER

Location Deception in Ad Hoc Wireless Networks

Autor: Jaime Mohedano Aragon

Director: Dr. Gang Qu

Madrid

ENGAÑO DE LOCALIZACIÓN EN REDES INALÁMBRICAS AD HOC

Autor: Mohedano Aragón, Jaime
Director: Dr. Gang Qu
Entidad Colaboradora: Universidad de Maryland

RESUMEN DEL PROYECTO

Este proyecto desarrolla simulaciones de red en ns-3 para mejorar la privacidad de localización en redes inalámbricas de sensores a través de seguridad en la capa física. El objetivo es dificultar que adversarios, estáticos o móviles, infieran la posición de los nodos en diversos escenarios, para ello, se emplean fluctuaciones en la potencia de transmisión.

Palabras clave: redes inalámbricas de sensores, privacidad de localización, seguridad de capa física, IEEE 802.15.4, potencia de la señal recibida, fluctuaciones de potencia de transmisión.

1. Introducción

El rápido crecimiento de las comunicaciones inalámbricas, en particular del Internet de las Cosas (IoT por sus siglas en inglés) y de los sistemas embebidos, han habilitado el desarrollo de redes inalámbricas ad hoc para comunicaciones en tiempo real entre nodos distribuidos como son los sensores. Sin embargo, estas redes son vulnerables a amenazas de seguridad inherentes a las comunicaciones inalámbricas. Un atacante puede aprovechar el hecho de que se esté realizando una transmisión inalámbrica, independientemente del contenido de la misma, para tratar de inferir la localización de los nodos de la red. De esta forma, se atacaría directamente a los metadatos de las comunicaciones, como la potencia de señal recibida, sin enfrentarse a las protecciones tradicionales basadas en cifrado.

2. Definición del proyecto

El proyecto desarrolla un conjunto de escenarios simulados de redes inalámbricas de sensores que incluyen distintos tipos de atacantes con el objetivo de comparar el nivel de protección de la localización de los nodos de la red con y sin el uso de técnicas de defensa. Dichas técnicas se basan en fluctuaciones de la potencia de transmisión en los nodos con el propósito de dificultar que los atacantes, móviles o estáticos, puedan estimar correctamente la posición de los nodos utilizando la potencia de la señal recibida y el modelo de propagación en espacio libre.

3. Descripción del modelo/sistema/herramienta

Este proyecto utiliza el simulador de red ns-3 para evaluar técnicas de privacidad de la localización en redes inalámbricas de sensores (WSN). El sistema modela una WSN compuesta por nodos sensores, un nodo sumidero y nodos atacantes, cada uno con funciones y capacidades distintas:

- Nodos de la red:
    - Nodos sensores y nodo sumidero: siguen una pila de protocolos por capas que incluye las capas UDP, IPv6, 6LoWPAN, IEEE 802.15.4 MAC y PHY. Se comunican mediante transmisiones UDP periódicas de 50 bytes.
    - Nodos atacantes: restringidos a operar en las capas PHY y MAC, no pueden participar en el enrutamiento de la red para garantizar una escucha pasiva.
- Entorno de simulación:
    - La red legítima opera dentro de una cuadrícula de 20x20 metros, que se ajusta a los despliegues IEEE 802.15.4 de baja potencia del mundo real.
    - Los atacantes estáticos permanecen fijos en sus posiciones iniciales, mientras que los atacantes móviles se mueven en función de las variaciones de RSSI.
- Comportamiento de los atacantes y procesamiento de paquetes:
    - Los atacantes interceptan pasivamente los paquetes en la capa física, midiendo los valores RSSI para estimar las distancias utilizando el modelo de pérdida de trayectoria en espacio libre o FSPL por sus siglas en inglés.
    - En escenarios coordinados, con múltiples atacantes, los atacantes comparten estimaciones de distancia para calcular intersecciones de círculos (con dos atacantes) o trilateración (con tres atacantes).
    - Las modificaciones realizadas en el código fuente de ns-3 para el etiquetado de paquetes en la capa PHY del protocolo LR-WPAN, permiten a los atacantes rastrear a los transmisores identificando las fuentes de los paquetes.
- Mecanismo de engaño:
    - La potencia de transmisión fluctúa dinámicamente entre -15 dBm y -5 dBm, creando variaciones artificiales de la señal para engañar a los atacantes.
    - Esta contramedida interrumpe los intentos de localización basados en RSSI, aumentando la dificultad de los adversarios para determinar con precisión la posición de los nodos.

La Figure 1 muestra el flujo de trabajo del proceso de simulación y análisis. Se utilizan métodos de Monte Carlo para generar un gran número de escenarios de red aleatorios, que se ejecutan en el simulador de red ns-3. Durante la simulación, se generan *logs* para capturar métricas clave como RSSI, distancias y posiciones de nodos. Estos resultados se procesan con Python para realizar un análisis detallado de las métricas de los escenarios. Además, ns-3 guarda datos de eventos durante la simulación, que posteriormente pueden visualizarse con la herramienta NetAnim.

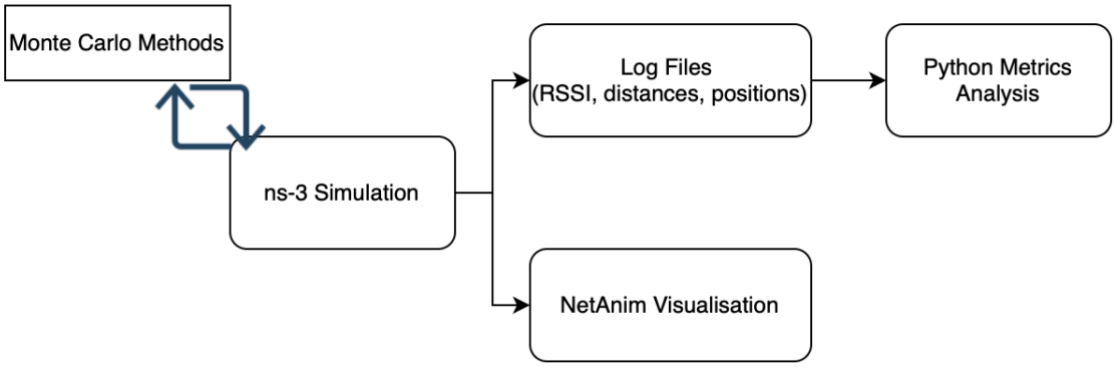


Figure 1. Arquitectura del sistema para simulación y análisis de WSN

4. Resultados

Los resultados demuestran la eficacia de los mecanismos de engaño a la hora de alterar la precisión de localización de los atacantes estáticos y móviles en una serie de escenarios. La Figure 2 presenta un diagrama de cajas comparativo de errores absolutos recortados, que expone tendencias claras en la propagación de errores. En escenarios con atacantes estáticos, los errores permanecen estrechamente distribuidos debido al impacto limitado de las fluctuaciones de energía en configuraciones no móviles. Sin embargo, a medida que aumentan la movilidad y la coordinación entre los atacantes, la distribución de los errores se amplía significativamente, reflejando la creciente influencia de las imprecisiones compuestas.

Los atacantes móviles, especialmente los que emplean la trilateración coordinada, experimentan errores de localización amplificados debido a que los niveles de potencia fluctuantes afectan a múltiples estimaciones de distancia simultáneamente. En el escenario de tres atacantes móviles, este efecto es más pronunciado, con valores extremos que indican casos en los que los mecanismos de engaño provocaron graves errores de localización. Estos resultados confirman que las estrategias de ataque más avanzadas no son intrínsecamente más resistentes, sino que, de hecho, se vuelven más vulnerables a los errores compuestos inducidos por el engaño, lo que las hace menos eficaces en el seguimiento del transmisor.

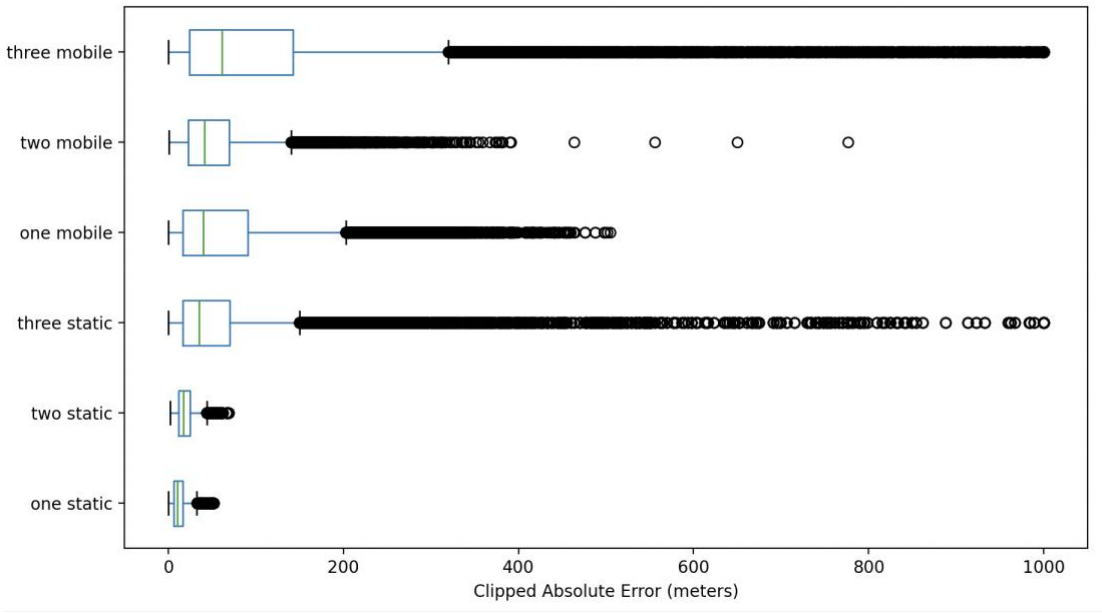


Figure 2. Diagrama de caja del error absoluto recortado por escenario

5. Conclusiones

Este estudio valida el potencial de los mecanismos de engaño, como las fluctuaciones de potencia, para proteger las redes de sensores inalámbricos contra atacantes cada vez más sofisticados. Mientras que los atacantes estáticos demuestran una relativa resistencia, los atacantes móviles coordinados sufren errores compuestos, en los que las sucesivas estimaciones incorrectas de distancia refuerzan los errores de cálculo, dando lugar a graves imprecisiones de localización.

Los conocimientos adquiridos sientan las bases para mejorar las defensas de la red mediante estrategias de engaño adaptativas y optimizadas, ofreciendo soluciones escalables para proteger los futuros sistemas de comunicación inalámbrica.

LOCATION DECEPTION IN AD HOC WIRELESS NETWORKS

Author: Mohedano Aragón, Jaime
Supervisor: Dr. Gang Qu
Collaborating Entity: University of Maryland

ABSTRACT

This project develops simulations in the ns-3 network simulator to enhance location privacy in wireless sensor networks through physical layer security techniques. By introducing fluctuations in the transmission power, it aims to mislead adversaries, both static and mobile, to infer the position of the nodes across several scenarios.

Keywords: Wireless Sensor Networks (WSN), location deception, physical layer security, IEEE 802.15.4, Received Signal Strength Indicator (RSSI), transmission power fluctuations.

1. Introduction

The rapid growth of wireless communications, particularly the Internet of Things (IoT) and embedded systems, has enabled the development of wireless ad hoc networks for real-time communications between distributed nodes such as sensors. However, these networks are vulnerable to security threats inherent to wireless communications. An attacker can take advantage of the fact that a wireless transmission is taking place, regardless of the content of the transmission, to try to infer the location of the nodes in the network. This would directly attack communications metadata, such as received signal strength, without dealing with traditional encryption-based protections.

2. Project definition

The project develops a set of simulated scenarios of wireless sensor networks that include different types of attackers in order to compare the level of protection of the location of the nodes with and without the use of defense techniques. These techniques are based on fluctuations of the transmission power at the nodes with the goal of making it difficult for attackers, mobile or static, to correctly estimate the position of the nodes using the received signal power and the free space propagation model.

3. Model/system/tool description

This project utilizes the ns-3 network simulator to evaluate location privacy techniques in WSNs. The system models a WSN consisting of sensor nodes, a sink node, and attacker nodes, each with distinct roles and capabilities.

- Network nodes:
    - Sensor nodes and sink node: they follow a layered protocol stack that includes UDP, IPv6, 6LoWPAN, IEEE 802.15.4 MAC and PHY layers. They communicate using periodic 50-byte UDP transmissions.
    - Attacker nodes: restricted to operating at the PHY and MAC layers, attackers cannot participate in network routing, ensuring passive eavesdropping.

- Simulation environment:
    - The legitimate network operates within a 20x20 meter grid, which aligns with real-world low-power IEEE 802.15.4 deployments.
    - Static attackers remain fixed at their initial positions, while mobile attackers move dynamically based on RSSI variations.
- Attacker behavior and packet processing:
    - Attackers passively intercept packets at the physical layer, measuring RSSI values to estimate distances using a Free-Space Path Loss Model.
    - In coordinated scenarios, i.e., scenarios with multiple atackers, adversaries share distance estimates to compute circle intersections (with two attackers) or trilateration (with three attackers).
    - Packet tagging modifications in ns-3's LR-WPAN PHY layer enable attackers to track transmitters by identifying packet sources.
- Deception mechanism:
    - Transmission power fluctuates dynamically between -15 dBm and -5 dBm, creating artificial signal variations to mislead attackers.
    - This countermeasure disrupts RSSI-based localization attempts, increasing the difficulty for adversaries to accurately determine node positions.

Figure 3 illustrates the workflow of the simulation and analysis process. Monte Carlo methods are used to generate a large number of randomized network scenarios, which are executed in the ns-3 network simulator. During the simulation, log files are generated to capture key metrics such as RSSI, distances, and node positions. These outputs are then processed using Python for detailed metrics analysis. Additionally, ns-3 saves event data during the simulation, which can later be visualized using NetAnim.

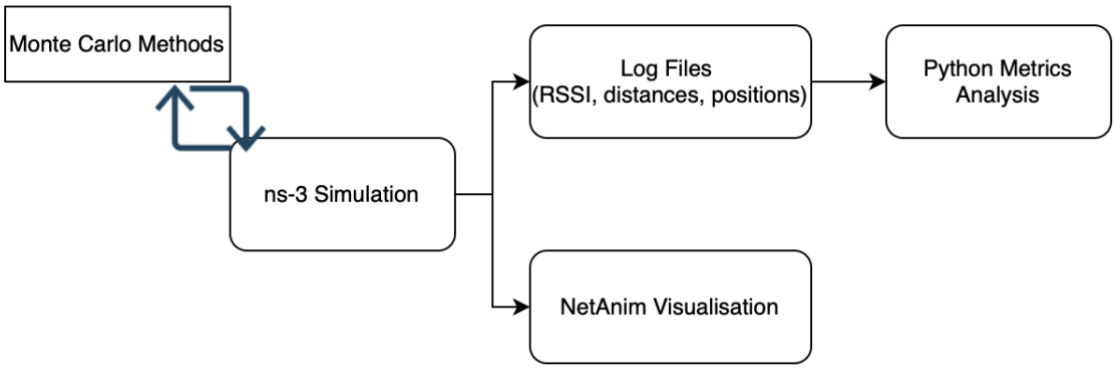


Figure 3. System Architecture for WSN Simulation and Analysis

4. Results

The results demonstrate the effectiveness of deception mechanisms in disrupting the localization accuracy of both static and mobile attackers across a range of scenarios. Figure 4 presents a comparative box plot of clipped absolute errors, highlights clear trends in error propagation. In scenarios with static attackers, errors remain tightly distributed due to the limited impact of power fluctuations on non-mobile setups. However, as mobility and coordination among attackers increase, the error distributions widen significantly, reflecting the growing influence of compounded inaccuracies.

Mobile attackers, particularly those employing coordinated trilateration, experience amplified localization errors due to fluctuating power levels affecting multiple distance estimates simultaneously. In the three mobile attackers scenario, this effect is most pronounced, with extreme outliers indicating cases where deception mechanisms led to severe mislocalizations. These findings confirm that more advanced attacker strategies are not inherently more resilient but, in fact, become more vulnerable to deception-induced compounding errors, making them less effective in tracking the transmitter.

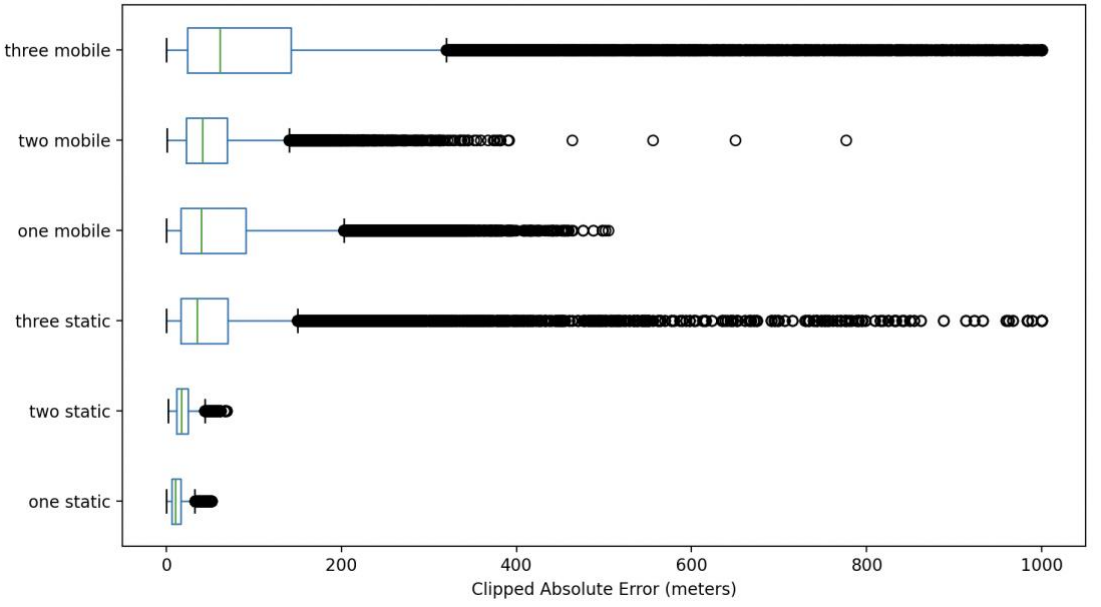


Figure 4. Box plot of clipped absolute error by scenario

5. Conclusions

This study validates the potential of deception mechanisms, such as power fluctuations, in protecting wireless sensor networks against increasingly sophisticated attackers. While static attackers demonstrate relative resilience, mobile coordinated attackers suffer from compounding errors, where successive incorrect distance estimations reinforce miscalculations, leading to severe localization inaccuracies.

The insights gained lay the foundation for enhancing network defenses through adaptive and optimized deception strategies, offering scalable solutions for securing future wireless communication systems.

Table of Contents

List of Figures

List of Tables

1. INTRODUCTION

The rapid advancement of wireless communication technologies, particularly the Internet of Things (IoT) and embedded systems, has enabled the creation of ad hoc wireless networks that facilitate real-time communication and data exchange between distributed nodes. These networks, which include Wireless Sensor Networks (WSNs) and other ad hoc systems, rely on a variety of devices such as IoT sensors and nodes, commonly tasked with critical functions such as environmental monitoring, smart infrastructure control, and coordination or navigation services. According to IoT Analytics' *State of IoT Summer 2024* report [1], the global number of connected IoT devices is projected to grow by 13% in 2024, reaching 18.8 billion by year-end. This growth highlights the increasing reliance on IoT systems across industries, with forecasts suggesting that IoT device installations will surpass 29 billion globally by 2027.

At the same time, Zscaler's research [2] highlights an 18% increase in IoT device traffic and the growing prevalence of malware attacks targeting IoT devices, reflecting the speed with which cybercriminals are exploiting the growing IoT landscape. IoT-related breaches have already led to significant consequences, with 34% of affected organizations reporting cumulative losses ranging from $5 million to $10 million [3]. These challenges underscore the critical need to protect IoT infrastructures from sophisticated threats.

Despite their utility, these networks face significant threats to the privacy of nodes at the physical layer. Unlike higher-layer attacks, which focus on content or protocol exploits, attackers who simply listen to signals, even if they are encrypted, can gain information about the position of nodes by measuring signal strength (RSSI), timing, or transmission patterns. This type of passive eavesdropping is often the first phase of a broader attack, in which an adversary performs reconnaissance to map network topology or track specific targets. Since the broadcast nature of wireless transmissions is inherently difficult to conceal at the physical layer, higher-layer solutions, such as data encryption or identity obfuscation, do not address vulnerabilities inherent in the characteristics of signal transmissions at the physical layer.

Conventional security mechanisms, while effective at protecting packet content, overlook how signal metadata can be exploited. Methods such as honeypots, trapdoor deployments, and mix-zones are highly application-specific or protect identity but not location. Similarly, cryptographic techniques, which focus on data confidentiality, integrity, or authentication, are ineffective against adversaries who analyze the properties of the signal itself, specifically "when" and "how strongly" a packet is transmitted, rather than "what" it contains. Thus, a malicious actor can deduce the approximate position of a node by analyzing RSSI or timings, completely bypassing common cryptographic defenses.

Motivated by these limitations, this project seeks to disrupt adversary reconnaissance efforts at the physical layer, hindering the extraction of meaningful positional data. Specifically, we investigate deception strategies designed to induce attackers to make incorrect or highly uncertain inferences about node locations. Rather than attempting to completely obscure the communications, a virtually impossible task given the fundamental broadcast nature of wireless transmissions at the physical layer, we propose to alter or randomize the characteristics of these transmissions. By maintaining a signal power level that is strong enough for legitimate receivers to decode while introducing fluctuations within regulatory limits, adversaries will find it more challenging to accurately infer node locations.

The relevance of this research increases due to the increasing dependence on ad hoc wireless networks in mission-critical scenarios. Industries and utilities rely on WSNs for a variety of applications, from environmental monitoring and industrial automation to disaster management and military defense. In all of these cases, protecting node locations is often paramount. By making reconnaissance at the physical layer significantly less reliable, overall network resilience is increased. Attackers who cannot confidently locate nodes face significant obstacles to planning subsequent attacks, such as targeted interference, man-in-the-middle infiltrations of specific nodes or physical tampering.

2. DESCRIPTION OF TECHNOLOGIES

2.1 NS-3 NETWORK SIMULATOR

ns-3 is an open-source and discrete-event network simulator targeted primarily for research and educational use [4]. The project began its development in July 2006 with the objective of building a replacement for its predecessor, ns-2. Built primarily in C++, ns-3 is structured as a set of shared libraries that users interact with through executables to define simulation topologies and configurations. It also allows users to program the simulation scripts in Python thanks to some Python bindings, offered via cppyy. The simulator is accessible across various operating systems like Linux, macOS, Windows, and FreeBSD.

The primary aim of ns-3 is to provide a free, user-friendly, and highly capable simulation tool for modern networking research [5]. It includes models for both IP and non-IP networks and is especially suited for wireless/IP simulations covering technologies such as Wi-Fi, LTE, and LR-WPAN. Researchers can use ns-3 to analyze network performance under conditions like interference, mobility, and congestion. Its real-time scheduling feature allows for simulation-in-the-loop scenarios, where ns-3 can interact directly with live systems.

It has a modular design which helps organize functionalities into protocol-specific modules and enables users to create simulations of varying scope and complexity. It offers detailed metrics, such as throughput, delay, and packet loss, which are particularly critical for projects like ours that require precise physical-layer manipulations, such as adjusting transmission power or timing. The simulator aligns closely with real-world protocols and allows the reuse of real application code, which facilitates accurate modeling and simplifies transitioning designs into practical deployments.

It was chosen for this project due to its modularity, extensibility, and emphasis on validated, reproducible research, as well as its active forum, where developers and users provide prompt and insightful support.

2.2 NETANIM

NetAnim is an offline graphical animator included with ns-3 and built on the Qt toolkit. It is designed to provide dynamic, time-stepped visualizations of the ns-3 simulation results by using XML-based trace files that are generated automatically during the ns-3 execution [6]. While NetAnim has not been actively maintained or updated since 2017, it remains a valuable resource for visualizing node movements, data exchanges, and topology changes in ns-3 simulations.

NetAnim reads trace logs and transforms them into a visual representation of how nodes interact over time. During playback, it highlights transmissions, displays logged packet metadata, and provides insights into network dynamics. This feature helps researchers validate whether simulated behaviors align with theoretical models.

For this project, NetAnim is particularly useful in scenarios involving mobile attackers. Visualizing logs in these cases can be cumbersome, but NetAnim simplifies the process by clearly illustrating attacker mobility and their interactions with network nodes. Additionally, it aids in assessing how proposed physical-layer modifications, such as signal strength fluctuations, affect the simulation. These visualizations make it easier to analyze and confirm the impact of such changes on network behavior.

2.3 IEEE 802.15.4 STANDARD

The IEEE 802.15.4 standard [7] defines the operation of low-rate wireless personal area networks (LR-WPANs) at the physical (PHY) and medium access control (MAC) layers. It is specifically designed for short-range, low power, low-data-rate communications and it is targeted at devices that are typically powered by batteries or resource-constrained. The standard was introduced in 2003 by the IEEE 802.15 working group and aims to enable cost-effective and energy-efficient wireless communications, being crucial for the Internet of Things (IoT) [8].

At the physical layer, IEEE 802.15.4 supports multiple ISM frequency bands, ranging from 2.4GHz to sub-GHz, with data rates ranging from 20 to 250 kbps. Its limited performance is intentional as it optimizes power efficiency for deployments such as sensor networks. The PHY layer specifies modulation techniques such as Direct Sequence Spread Spectrum (DSSS) and includes features such as power sensing, Clear Channel Assessment (CCA) and Link Quality Indicator (LQI) [9].

The MAC layer provides mechanisms for coordinating how devices share the wireless medium. Its primary access method is carrier sense multiple access with collision avoidance (CSMA/CA), which ensures efficient channel usage by detecting free channels and reducing interference. For priority or time-sensitive data, the standard includes guaranteed time slots (GTS) to reserve bandwidth. Other MAC functionalities include synchronization, beaconing, and secure communication through 128-bit AES encryption.

In terms of the network model, IEEE 802.15.4 supports star, point-to-point, and mesh topologies. Star networks are centralized around a coordinator, while point-to-point and mesh topologies allow data to be routed through intermediate nodes, enabling communication over extended distances. Nodes can take on one of two network roles: full-function devices (FFDs), which can act as coordinators or routers, and reduced-function devices (RFDs), which are extremely simple and designed for basic operations, only able to communicate with FFDs.

As for the advantages and disadvantages of the technology; the standard's low cost, power efficiency, and minimal overhead characteristics make it ideal for applications requiring long battery life, such as industrial sensors, home automation systems, and remote controls. However, limitations such as low data rates, potential interference (particularly in the 2.4 GHz band), and occasional latency in contention-based systems are its main disadvantages. Despite these challenges, IEEE 802.15.4 provides the basis for other higher-level protocols such as ZigBee, 6LoWPAN, WirelessHART, and Thread, which improve its capabilities for routing, networking, and application-layer services.

The choice to use IEEE 802.15.4 over alternatives like Wi-Fi is driven by its closer alignment with the characteristics of most real-world WSNs. These networks often emphasize low power consumption and limited data rates, making IEEE 802.15.4 a better fit due to its focus on energy efficiency and low-bandwidth communication. In contrast, Wi-Fi is designed for higher power and throughput, which are less suitable for scenarios where battery life and resource constraints are critical.

2.4 6LoWPAN

6LoWPAN, which stands for "IPv6 over Low-Power Wireless Personal Area Networks" [10], is a protocol adaptation layer designed to enable IPv6 connectivity over low-power, resource-constrained wireless networks, particularly those based on the IEEE 802.15.4 standard. Developed by the IETF's 6LoWPAN working group, its primary goal is to allow small, low-power devices to connect to IP-based networks. It has been widely adopted in IoT applications, including industrial automation, smart metering, agriculture, and building automation. It simplifies network infrastructure by enabling direct communication with cloud services due to natively supporting IPv6 [11].

One of the main design challenges 6LoWPAN addresses is the mismatch between the large packet sizes required by IPv6, it requires the Maximum Transmission Unit (MTU) to be at least 1280 bytes, and the much smaller frame sizes supported by IEEE 802.15.4, whose frames are limited to 127 bytes, with even less room for higher-layer data after accounting for overhead. To resolve this, 6LoWPAN introduces an adaptation layer that handles packet fragmentation and reassembly, allowing IPv6 datagrams to be divided into smaller fragments and reassembled at their destination. Updates such as RFC 6282 further enhance this by compressing IPv6 headers, reducing the packet overhead and optimizing performance for constrained devices [12].

6LoWPAN's addressing system integrates with IEEE 802.15.4's use of 64-bit extended addresses or 16-bit short addresses assigned within a Personal Area Network (PAN). To conserve energy in low-power devices, it optimizes IPv6 neighbour discovery, as defined in

RFC 6775, minimizing unnecessary broadcast traffic. The protocol also allows data to travel across intermediate nodes to reach gateways or final destinations using multi-hop mesh routing.

Security in 6LoWPAN is achieved through a layered approach. As it has been mentioned in the previous technology, at the link layer, IEEE 802.15.4 provides AES-128 encryption for secure data transmission. Higher layers can implement end-to-end security protocols like DTLS or TLS, which ensures that data remains protected both within the local network and during transmission across the internet.

6LoWPAN was chosen because it builds on the low-power, short-range capabilities of IEEE 802.15.4 while enabling integration with full IP networking. Since IEEE 802.15.4 aligns well with the energy limitations and modest data rates typical of real-world sensor networks, the next step was a way to connect those nodes directly to standard internet protocols. 6LoWPAN solves this challenge by compressing and fragmenting IPv6 packets, allowing resource-constrained devices to operate using “native” IP, bridging the gap between local sensors and the larger cloud-connected IoT infrastructure.

3. STATE OF THE ART

Existing research in wireless sensor networks (WSNs) and low-rate, low-power protocols such as IEEE 802.15.4 and 6LoWPAN has addressed security mainly in terms of confidentiality, integrity and authentication. While these approaches prevent attackers from reading message contents, it does not address the problem of location privacy in which adversaries attempt to deduce node positions by analyzing transmission patterns, signal strengths, and routing information.

1. Deception-based traffic analysis countermeasures

Ebrahimi and Younis [13] proposed the Assisted Deception (AD) model to counter powerful adversaries who perform temporal and spatial correlations to attack WSNs. Conventional security measures often fail to prevent these types of pattern-inferring attacks because they focus primarily on protecting data confidentiality and integrity. The authors identified a significant threat posed by enhanced evidence theory (EET), an advanced traffic analysis model that combines spatial and temporal correlations to improve adversarial accuracy in locating critical nodes, such as base stations. EET exploits the timing and frequency of packet relays, enhancing the attacker's ability to reconstruct network topologies.

The AD model counters this by leveraging a distributed, cooperative strategy among nodes to inject deceptive packets at carefully timed intervals. These packets can be routed through the network to create false traffic patterns or remain isolated to disrupt adversarial observations. This approach effectively obscures the temporal correlation between legitimate transmissions, making it challenging for adversaries to establish meaningful links between observed data flows. One of the advantages of the AD model compared to centralized or static methods is that it dynamically adapts to both event-driven and time-driven network operations with minimal overhead. Simulations demonstrate that the model significantly reduces an attacker's ability to deduce node positions while maintaining network efficiency.

2. SDN-based topology deception in UAV networks

Tan et al. [14] introduced a Software-Defined Networking (SDN)-based virtual topology mechanism aimed at misleading attackers in unmanned aerial vehicle (UAV) networks with the objective of mitigating the impact of topology probing based targeted attacks. These networks rely on key drones with critical roles, such as acting as data relays or hosting essential service, whose recognition and disruption, as highlighted by the study, could have catastrophic consequences. To avoid this, the authors propose the use of honeypot drones that imitate the functionality of critical drones, i.e., deployed and acting as decoys.

This mechanism leverages the centralized control capabilities of an SDN to construct virtual dynamic network topologies that redirect attackers to honeypot drones. The attacker is mislead by the decoy nodes, which appear as high-priority targets, drawing attention away from the actual key drones, a graphical example is shown in Figure 5. The simulations performed in the study show significant reductions in connectivity losses even under scenarios where a substantial portion of drones are targeted.

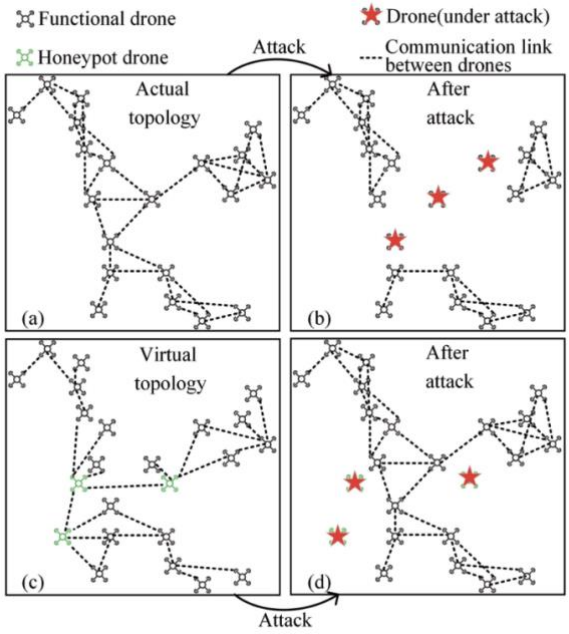


Figure 5. Graphical example of the protection of key drones, retrieved from [14]

3. Trap deployment for anti-eavesdropping

He et al. [15] designed a proactive trap deployment system aimed at enhancing security in wireless networks against eavesdropping attacks. The system addresses the vulnerabilities posed by the inherent broadcast nature of wireless communications, which makes them susceptible to interception by adversaries. The novel aspect about this approach lies in its combination of physical-layer randomization with the strategic use of fake data transmissions. This combination ends up confusing a potential eavesdropper and safeguarding legitimate communications. The system leverages MU-MIMO (Multi-User Multiple Input Multiple Output) technology to enable the simultaneous delivery of true messages to legitimate users and fake messages to adversaries.

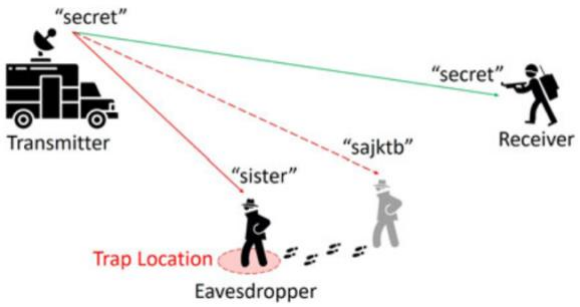


Figure 6. Sending a deceptive command to the eavesdropper, retrieved from [15]

The paper details a double strategy: first, to lure eavesdroppers to a designated "trap region" by transmitting signals that appear meaningful but are deceptive; and second, to maintain secure and unobservable communication channels between legitimate nodes. This trap region is dynamically configured using multiple antennas, ensuring that adversaries approaching it receive increasingly clear but fake messages, while legitimate users remain unaffected as shown in Figure 6. Experimental results using Universal Software Defined Radio Peripherals (USRPs) validate the system's effectiveness, demonstrating how the deliberate manipulation of Signal-to-Noise Ratios (SNR) can guide eavesdroppers and protect sensitive communications.

4. Encryption and fake identity in IoBT networks

Alkanjr and Mahgoub [16] presented a strategy to protect location privacy in Internet of Battlefield Things (IoBT) networks by employing deception-based techniques. The proposed scheme combines encryption methods, dummy IDs, and dummy packets to obscure real node positions and confuse adversaries attempting to map network topology.

The approach involves each node sending out both real packets containing actual location information and fake packets with dummy data. This method confuses attackers trying to determine the true positions of military assets. The use of dummy IDs adds another layer of deception, making it harder for adversaries to identify the real nodes. Encryption further ensures that even if communication is intercepted, its content remains secure.

The authors evaluated their scheme using a mathematical model to measure key metrics, including safety time (ST), probability of failure (PF), and the probability of identifying real packets (PIRP). Simulations conducted in NetLogo under different mobility models, such as wiggle and straight-line movement, confirmed the efficacy of the approach. The results demonstrated that the scheme significantly improved location privacy by reducing the probability of adversaries successfully identifying real nodes.

5. Expandable mix-zones for location privacy

Butun and Mahgoub [17] extended the mix-zone concept, originally for IoT, into the military context of the Internet of Battlefield Things (IoBT). Their approach introduces “Expandable Mix-Zones,” dynamic zones that obfuscate the precise locations of users, such as soldiers or IoBT-enabled mobile units, while still enabling essential location-based services (LBS) to function, a graphical example is shown in Figure 7. Traditional static mix-zones provide fixed geographical areas for anonymizing user presence, on the other hand, expandable mix-zones adapt their size dynamically based on operational requirements, expanding or contracting as needed to enhance privacy. Within these zones, LBS only receives binary

information about user presence wthout any precise location data, reducing the risk of adversaries tracking locations.

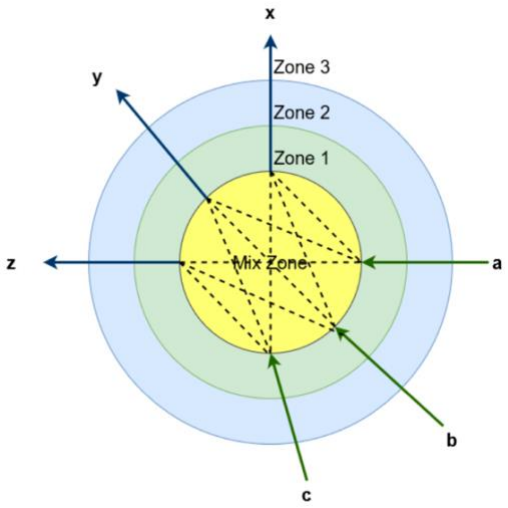


Figure 7. An illustration of the expandable mix-zone concept for IoBT, retrieved from [17]

The study demonstrated that by strategically deploying expandable mix-zones in sensitive battlefield areas, such as military bases or conflict zones, it becomes possible to safeguard critical location information. Users entering these zones exchange pseudonyms, ensuring anonymity and making it difficult for adversaries to correlate entry and exit points. However, the paper notes that the success of the method depends on user cooperation and robust pseudonym exchange mechanisms.

6. Physical-Layer Security Mechanisms for Resource-Constrained WSNs

Choi, Ha and Jeon [18] adressed security challenges in WSNs by leveraging physical-layer properties. Traditional cryptographic techniques are effective for securing data but are often impractical for WSNs because of the power constraints of sensor nodes. To tackle this, the authors opted for physical-layer security schemes as a lightweight solution, leveraging some inherent wireless channel properties like randomness and signal-to-noise ratio gaps.

One of the main points of the paper is the use of distributed detection in WSNs, where multiple sensors relay their observations to a fusion center for collective decision-making. The study propose methods to secure this communications between the sensors and the fusion center while spoiling eavesdropping attempts from an adversary refered to as the Enemy Fusion Center (EFC). Two key techniques discussed are stochastic encryption and channel-aware encryption. Stochastic encryption introduces random noise into sensor transmissions to degrade the eavesdropper's ability to infer accurate information while channel-aware encryption exploits the natural randomness of wireless channels to create encryption keys dynamically. This approach guarantees that the lack of shared channel information makes the data incomprehensible even if the signals are intercepted.

7. Enhancing Wireless Sensor Network Security through Opportunistic Scheduling

Marjanović et al. [19] proposed an opportunistic scheduling mechanism to enhance the physical-layer security in resource-constrained environments as WSNs, the study focuses on leveraging wireless channel characteristics to secure data transmission against eavesdroppers.

The proposed method is based on opportunistic scheduling, where only the sensor node with the best instantaneous channel conditions is allowed to transmit data. This selection process reduces the likelihood of successful eavesdropping by adversaries since the opportunistic scheduler prioritizes the most favorable channels for legitimate communication. The method takes advantage of the inherent randomness and variability of wireless channels and enhances the secrecy capacity, defined as the maximum rate at which data can be securely transmitted without being intercepted by eavesdroppers.

Simulation results validate that the proposed method significantly improves the security of WSNs in scenarios with multiple sensors and eavesdroppers. The authors also highlight the method's efficiency in maintaining security without introducing significant computational overhead, making it particularly suitable for WSN deployments.

4. DESCRIPTION OF THE WORK

4.1 JUSTIFICATION

The motivation of this project originated from a military application, where troops depend on ad hoc wireless sensor networks (WSNs) for real-time situational awareness in the battlefield. In those environments, adversaries can exploit traffic patterns and signal strengths to extract valuable information, sometimes more effective than decrypting secured data because of an effort-to-result tradeoff. As the project progressed, its scope shifted to a broader approach since any Internet of Things (IoT) or embedded system operating on low-power ad hoc networks face similar risks. Existing solutions typically focus on protecting the contents of the communications, but they rarely address physical-layer characteristics that attackers could leverage to locate sensor nodes.

This project aims to bridge that gap by developing simulations in a network simulator that model a variety of scenarios involving different numbers of attackers and sensor nodes. The scenarios include both static and mobile attackers, allowing for comprehensive analysis of diverse adversarial behaviors. To evaluate the effectiveness of defense mechanisms, the same scenarios are simulated with and without the implementation of power fluctuation techniques as a defensive mechanism.

4.2 OBJETIVES

This project aims to address the security vulnerabilities in ad hoc wireless networks, particularly those involving adversaries exploiting the physical-layer characteristics of wireless transmissions to infer node positions. The following objectives guide the work:

1. Understand the problem and its origin

Part of the complexity of this project lies in the problem formulation, particularly the inherent physical characteristics of wireless transmissions such as the transmission power,

which cannot be cancelled. This objective includes a comprehensive review of relevant literature and analysis of high-level deception techniques previously proposed (not implemented) by the supervisor's research group to study their feasibility.

2. Design a power-fluctuation defense mechanism

Develop a protocol-level approach in which nodes introduce variations in their transmission power. The goal is to disrupt the attacker's ability to estimate the location of sensor nodes by signal-strength analysis. It includes determining how these power variations can be triggered or randomized without undermining network performance significantly.

3. Implement and simulate multiple scenarios in ns-3

To validate the proposed mechanism, this objective involves implementing and testing the designed techniques using the network simulator ns-3. Simulations will be conducted in randomly generated network environments with several variables such as number of attackers and their mobility (static/mobile), number of sensor nodes, etc. Incorporating both baseline (no defense) and defensive modes (power fluctuations) and measuring how effectively the attacker's location inference is degraded under each scenario.

4. Ensure practical applicability beyond simulations

While simulations provide a controlled environment with access to all packet layers and relatively ideal conditions, real-world challenges must be considered, such as adversaries using Software-Defined Radios (SDRs) to passively monitor the spectrum. This objective ensures the solutions address such complexities for real-world deployment. Also, documenting any constraints or assumptions made in the simulations that would need re-assessment prior to deploying such a deception strategy on physical IoT or sensor hardware.

4.3 WORKING METHODOLOGY

Since this project is mainly software-focused, the methodology chosen to be followed was the Agile methodology, which is one of the most appropriate and successful approaches in these cases. In Agile, the planning is incremental, several tasks are done in parallel, and it is easier to modify the process to reflect the changing needs of the client, supervisor in my case. The requirements and the pertinent solution evolve and change in time in relation to the client's needs.

All the tasks of the project are collected in the product backlog, these tasks or features are agreed to be done in certain periods of time, and each period is called a Sprint. Similar to the product backlog, the collection of tasks belonging to a Sprint is called the Sprint backlog. The Sprints divide the timeline of the project and guarantee that the client is constantly receiving partial but constant deliveries. At the beginning of each Sprint, a meeting to organize everything takes place, and, at end of each Sprint, the progress made is presented with the objective to do a Sprint review. In this case, I decided to follow a short-sprint approach, the duration of each sprint was two weeks, so, since the duration of the project was four months, there were eight sprints.

Some of the core values of the Agile Manifesto were followed:

- Working software over comprehensive documentation: prioritizing bi-weekly deliveries with working software to allow the supervisor to share possible additions to the simulations and to provide feedback.
- Responding to change over following a plan: a broad plan and clear objectives were established but circumstances evolved. For example, I started reviewing literature related to WSN and its security and started drafting ideas of a complex deception protocol which later proved to be too complicated to be implemented in the simulations and was not feasible for the timeframe of the project.

With these Agile principles and short sprints in mind, the project's tasks were organized into several phases reflecting deliverables. Below is an outline of these phases, followed by a Gantt chart to provide a visual roadmap of this plan.

1. Literature review and problem definition

The first step involves clearly defining the problem and its underlying assumptions. This requires reviewing the existing literature on WSN security, location inference attacks, and deception techniques and identifying gaps in current solutions. Different scenarios are considered, including single or multiple attackers, as well as static or mobile adversaries.

2. Theoretical solution design

With the problem and attacker behaviors defined, the next step is to develop a set of techniques to address all identified scenarios. These solutions should be both realistic and adaptable. The complexity of defense increases with attacker capabilities. For instance, in a scenario with three static attackers, the adversaries may use trilateration, intersecting three circles based on signal strength, to determine a node's location.

3. Simulation environment setup

The proposed techniques are implemented and evaluated using the ns-3 network simulator. A gradual approach is proposed: starting with basic WSN scenarios, followed by static attackers, multiple attackers, mobile attackers, and finally applying power fluctuation deception strategies. ns-3's flexibility allows testing under various topologies, densities, and attacker behaviors. Additionally, NetAnim provides visual insights into network dynamics, particularly useful in mobile attacker scenarios.

4. Experimental evaluation and analysis

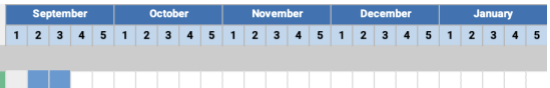
A series of simulations assess the effectiveness of the deception strategies. Scenarios feature static and mobile attackers, several numbers of attackers, and different network configurations, including random node placement. Key performance indicators (KPIs) and performance metrics are recorded, allowing a comparative analysis of scenarios with and without deception.

5. Iteration and refinement

Based on the simulation results, adjustments are made to improve protocol parameters and tactics. This may include optimizing power fluctuation patterns, refining timing intervals, or refining attacker positioning for more realistic conditions.

The following Gantt chart illustrates the timeline and progression of tasks throughout the project. Several delays occurred during the project's lifetime, including a one-week delay at the end of both (i) completing the ns-3 tutorials and (ii) reviewing the ns-3 protocol implementation, specifically with LR-WPAN physical and MAC layers. Additionally, a two-week arose during the implementation of multiple mobile attackers due to the need to carefully consider whether the attackers would act independently or coordinate their movements towards the nodes. Finally, another one-week delay occurred at the start of defining performance metrics. Despite the setbacks, the overall project timeline remained unaffected since the delays did not impact the progression of dependent tasks.

WBS NUMBER	TASK TITLE	START DATE	DUE DATE	PCT OF TASK COMPLETE
1	**Literature Review & Problem Definition**			
1,1	General WSN Characteristics	9/9/24	9/20/24	100%
1,2	WSN Localization Methods	9/16/24	10/27/24	100%
1,3	WSN Security	9/23/24	10/4/24	100%
1,4	Location Deception Techniques in WSNs (State of the Art)	10/4/24	10/25/24	100%
1,5	Define Problem Statement & Potential Solutions	10/4/24	10/25/24	100%
2	**Theoretical Solution Design**			
2,1	Develop Theoretical Solution Approaches (Power Fluctuations, Dummy Nodes, Geofencing)	10/4/24	10/25/24	100%
3	**Simulation Environment Preparation**			
3,1	Set Up & Configure ns-3 Simulation Environment	10/21/24	10/25/24	100%
3,2	Complete ns-3 Tutorials & Familiarization	10/21/24	10/27/24	100%
3,3	Review ns-3 Default Protocol Implementations	10/29/24	11/8/24	100%
3,4	Select & Validate Wireless Protocols	10/29/24	11/8/24	100%
3,5	Implement Physical Layer (PHY) Modifications	11/1/24	11/15/24	100%
4	**Test Scenario Development & Execution**			
4,1	Static-Attacker Test Scenarios (1, 2, 3 Attackers)	11/11/24	12/6/24	100%
4,2	Static-Attacker Test Scenarios with Deception (1, 2, 3 Attackers)	12/9/24	12/13/24	100%
4,3	Mobile-Attacker Test Scenarios (1 & 2 Attackers, Dependent/Independent)	11/25/24	12/31/24	100%
4,4	Mobile-Attacker Test Scenarios with Deception (1 & 2 Attackers, Dependent/Independent)	12/16/24	12/31/24	100%
5	**Final Improvements & Performance Evaluation**			
5,1	Generalize Scenarios into a Single Configurable Setup	12/16/24	12/26/24	100%
5,2	Refine Attacker Positioning for Realistic Conditions	12/26/24	1/10/25	100%
5,3	Define KPI's & Performance Graphs	11/25/24	1/31/25	100%



4.4 BUDGET ESTIMATE

No direct hardware or proprietary software expenses were incurred. The free open-source network simulator ns-3 and free libraries were utilized throughout the project. Thus, the principal resource investment was the developer's labor, estimated at 36 hours per week over 16 weeks, totalling in 576 hours.

5. PLATFORM CONFIGURATION AND SETUP

This section outlines the steps taken to prepare the ns-3 platform for running the simulations in this project, including installation, configuration of the environment and general usage. This project has been developed using macOS, thus, specific details may vary depending on the operating system and hardware used. However, the overall approach remains consistent with the official ns-3 documentation [20].

5.1 INSTALLING NS-3

Prerequisites

ns-3 has various optional extensions, but the main features just require a C++ compiler (g++ or clang++), Python (version 3.6 or above), CMake and a build-system (e.g. make, ninja, Xcode). Since, this project was developed on macOS, clang++ was used (available in Xcode or Xcode Command Line Tools). Details on the specific tool versions utilized during this project are available in the Annex II: Dev/Build Tool Information section.

Obtaining the Source Code

ns-3 is distributed in source code only. There are two main ways to obtain the source code:

1. Downloading the latest release as a source code archive from the main *ns-3* website
2. Cloning the Git repository from GitLab.com

For this project, option 1 was followed, and the source code was downloaded from the official releases' page [21]. It should be mentioned that the newest ns3 version when developing the project was 3.42 but it didn't support wireless packets for its corresponding NetAnim version, 3.108, so a downgrade was needed in order to visualize the graphical animations.

Configuring and Building ns-3

The next step is to configure the build using the *CMake* build system. The default setup is sufficient for most scenarios, enabling example programs and tests, with assertions and logging support activated. To configure and build ns-3:

```
cd ns-allinone-3.37/ns-3.37
./ns3 configure --enable-examples --enable-tests
./ns3 build
```

Testing the instalation

Once completed, verify the installation by running the ns3 equivalent to a "Hello, world!" program:

```
./ns3 run hello-simulator
```

With a run command ns3 checks to make sure that the program is built correctly and executes a build if required. Then, it executes the program, in this case it will output:

```
Hello Simulator
```

5.2 CONFIGURING THE SIMULATION ENVIRONMENT

The process of creating a simulation in ns-3 can be divided into several logical steps, which are presented below [4]:

1. Topology definition: setting up the fundamental structure such as number of nodes, their mobility and positions, how they are cononected. To ease this process and tu automate this creation, ns3 provides helper classes and containers. For example, the *MobilityHelper* can assign mobility models to nodes, allowing for static or mobile scenarios, or *NetDeviceContainer* that manages the assignment of network interfaces.

2. Model development: adding models to the simulation to simulate specific network behaviors such as protocol stacks (e.g. UDP, IPv4) or communication technologies (e.g.

point-to-point, IEEE 802.15.4). Most of the time this is done using helpers such as *Ipv4AddressHelper* or *LrWpanHelper* to ensure consistency in the configurations.

3. Node and link configuration: to reflect real-world parameters, each model and node in the simulation requires configuration. ns3 provides an attribute system to modify default settings such as such as transmission power, packet size, and data rate.

4. Execution: once the simulation is configured, ns-3 generates events that simulate packet transmissions, mobility and any interaction between nodes. These events are logged as the simulation time passes.

5. Performance analysis: after the simulation is finished, data is available as a time-stamped event trace. This data can then be statistically analysed with tools like R, Python or Matlab to derive metrics and draw conclusions.

6. Graphical Animation: data can be visualized using tools like NetAnim, matplotlib or Gnuplot. NetAnim provides a dynamic visualization of node interactions, including mobility patterns and packet transmissions. Other tools enable plotting key metrics.

This general workflow applies to most ns-3 simulations but specific configurations for each scenario in this project, such as number of nodes, attacker mobility patterns and more, are outlined in the following chapter for each scenario.

5.3 RUNNING SIMULATIONS AND VISUALIZING OUTPUTS

Once the simulation environment is configured, the next step is to execute the simulation and visualize the resulting data.

Executing Simulation Files

Simulations in ns-3 are executed through the ./ns3 command-line utility, which is part of the ns-3 framework. To run your specific simulation file, the syntax is as follows:

```
./ns3 run "exampleScenario"
```

The example-scenario.cc file must be in the scratch directory in order to be automatically built if you run the ns3 command.

Command-line parameteres can be added to the simulation, which is particularly useful for performing batch runs or Monte Carlo simulations. These parameters allow for dynamic configuration of aspects such as the number of nodes, attacker behavior, or mobility models. The format to be used is as follows:

```
./ns3 run "exampleScenario --param1=value1 --param2=value2"
```

Visualizing outputs

To analyze the behavior of simulations, ns3 provides multiple mechanisms, from text-based logs to graphical tools.

- **Text-based logs**

The ns-3 logging system is used to debug and analyze simulations, it allows to monitor simulations with fine-grained control by enabling certain log components. Using the **NS_LOG_COMPONENT_DEFINE** macro, a logging component is defined, and logs can be activated at various verbosity levels (e.g., DEBUG, INFO, or WARN).

- **TraceSources**

TraceSources are crucial for ns-3's event-driven architecture since they offer a structured way to capture simulated data. They trigger signals when specific events occur, such as packet transmission or reception. Users can connect custom callbacks to TraceSources, enabling real-time data collection and processing. For example, tracing the throughput of a wireless link can be achieved by subscribing to the appropriate TraceSource.

- **Trace Files**

ns-3 also allow for the creation of trace files that capture specific metrics, such as packet drops, delays, or throughput. These files are often exported in .pcap or .txt formats and can be analyzed using external tools for statistical analysis.

- **NetAnim visualization**

For visualizing network dynamics, ns-3 provides NetAnim, which is a really powerful and intuitive tool. To install NetAnim on macOS (using brew to install Qt):

```
brew install qt@5
cd netanim-3.108
make clean
qmake NetAnim.pro
make
```

For additional installation guidance, refer to the NetAnim wiki [6].

To use NetAnim:

1. Include the NetAnim module header in your simulation script:

```
#include "ns3/netanim-module.h"
```

2. Enable animation traces before calling *Simulator::Run()*:

```
AnimationInterface anim("simulationOutput.xml");
```

3. Run NetAnim and open the generated XML file using the file selection icon in the top-left corner:

```
./NetAnim
```

In this project, .txt and .pcap log files were particularly useful for analyzing scenarios with static attackers. However, for scenarios involving mobile attackers, NetAnim proved to be more useful due to its ability to visually track attacker mobility across the simulation grid.

5.4 PROTOCOL CONSIDERATIONS

Since this project focuses on the security of IEEE 802.15.4 at the physical layer, it is important to outline some key physical layer properties. While the standard allows for several configurations, the simulations adopt realistic values that align with real-world implementations and are consistent with ns-3's implementation, these are shown in Table 1.

Table 1. Physical layer properties for IEEE 802.15.4 used in simulations

Property	Value
Modulation	O-QPSK
Frequency Band	2.4 GHz ISM band, using Channel 11 at 2405 MHz
Max PSDU Size	127 octets (maximum PHY Service Data Unit)
Data Rate	250 kbps (4 bits per symbol, 62.5 ksymbols/s)
Default Tx Power	0 dBm (1 mW)
Rx Sensitivity	-106.58 dBm
Propagation Loss Model	Log-Distance (Path loss exponent = 3.0)
Noise Power	-106.987 dBm (Noise factor 1.0)
Channel Bandwidth	2 MHz
Interference Considerations	Uses DSSS to mitigate Wi-Fi/Bluetooth interference

6. SYSTEM DEVELOPED

This section describes the implementation of the simulation environment and the attacker scenarios used to evaluate RSSI-based localization under transmission power fluctuations. The simulations are divided into static and mobile attacker scenarios, each with variations in the number of attackers and their coordination strategies. Static attackers remain fixed and estimate distances from a single position, while mobile attackers adapt their movement based on RSSI trends.

6.1 OVERVIEW

The primary goal of these simulations is to analyze the impact of transmission power fluctuations on an attacker's ability to estimate distances and node locations based on RSSI values. The simulations are divided into static and mobile attacker scenarios, with variations in the number of attackers and their level of coordination.

Table 2. Protocol stack installed on network nodes

UDP
IPv6
Adaptation Layer (6LoWPAN)
IEEE 802.15.4 MAC
IEEE 802.15.4 PHY

The network consists of three different types of nodes: sensor nodes, a sink node and one or more attacker nodes. The nodes that belong to the WSN, i.e. the sensor nodes and the sink

node, have the protocol stack from Table 2 installed. In contrast, attacker nodes only have the PHY+MAC 802.15.4 layers to prevent them from participating in routing and communication mechanisms, which would not occur in a real-world scenario. The sensor nodes act as UDP clients, transmitting 50-byte packets every second, while the sink node function as both the UDP server and the PAN coordinator, handling all the network control operations.

The initial positioning occurs within a 20x20 meter grid as in a real-world implementation of low-power IEEE 802.15.4 networks. Since these networks are usually deployed in shor-range scenarios, this range limitation aligns well with real deployments. In mobile attacker scenarios, attackers can move outside of this grid based on their estimation of the transmitter's opsition, while in static scenarios, the attackers remain fixed in place.

Each simulation runs for a total of 30 seconds, during which attackers intercept packets, measure RSSI values, and attempt to estimate transmitter locations. This duration was chosen to ensure sufficient packet exchanges for meaningful results while maintaining computational efficiency.

Attacker Behaviour and Packet Processing

Attacker nodes are passively listening to the wireless communications at the physical layer, intercepting packets and measuring RSSI values. Since IEEE 802.15.4 is a wireless protocol and, thus, has a broadcast nature, attackers can receive and analyze packets without disrupting normal network operations. With the RSSI value and some other predefined parameters of the Propagation Loss Model, attackers can compute an estimate of the distance to the transmitting node.

> *It is important to note that in real-world deployments, these assumed parameters (antennas' gains, path loss exponent, etc) may differ, leading to inaccuracies in estimation,* ***the purpose of this project is to compare scenarios with power fluctuations and without them to prove its performance, that is why the propagation model parameters do not change****.*

In real-world implementations, attackers would typically perform frequency sweeps to identify the WSN's operating channel before attempting to capture packets. Additionally, in scenarios with coordinated attackers, they must establish a secure communication channel without interferring with the WSN's transmissions. However, in the simulation environment, several modifications were made to simplify implementation while preserving the core attack principles. As previously mentioned, attackers are restricted to LR-WPAN layers to ensure a purely passive role.

Packet Identification Challenges at the PHY Layer

One key challenge in the simulation setup is packet tracking as the physicial layer does not inherently provide identification of the nodes, neither the sender nor the receiver. In ns-3, protocols are installed on nodes through predefined models as configured in the source code and modifications can be made to this protocols to suit specific research needs by recompiling the simulator code. Since the objective was to made attackers work at the PHY layer, a custom tagging mechanism was implemented in ns-3's LR-WPAN PHY layer to identify the source and destination of every packet, which allows better logging and analysis.

Simulation Considerations and Deception Mechanism

Although the receiver sentivity implemented in ns-3 for IEEE 802.15.4 is -106.58 dBm, the simulation is configured to allow all packets to be received, regardless of the RSSI value. This ensures that transmission power fluctuations can be effectively studied and that mobile attackers can estimate distances even outside the theoretical communication range.

Additonally, for all scenarios, the network topologies are randomly generated to allow for Monte Carlo simulations. The script used to randomly generate these network topologies for Monte Carlo simulations is included in Annex III: Monte Carlo Method Script for Random Network Topology Generation for reference. This ensures a wide variety of spatial configurations and provides a robust evaluation of the system's performance across diverse real-world-like deployments.

In scenarios with deception, sensor nodes and the sink node dynamically adjust their transmission power, by default in normal transmissions it is set to 0 dBm, in deception scenarios they linearly change between -15 dBm and -5 dBm, creating an artificial movement effect in order to mislead attackers. This causes them to miscalculate distances, making the node localization even more challenging.

Finally, there is a key distinction between scenarios with static and mobile attackers. In static scenarios, attackers remain fixed at their initial positions and rely only on RSSI-based distance estimation to infer the transmitter location. On the other hand, in mobile attacker scenarios, adversaries adjust their position dynamically based on RSSI changes. If an attacker detects an increase on the RSSI, it continues moving in the same direction assuming it is approaching the target. If the RSSI decreases or stays the same, the attacker randomly changes direction, searching for an alterntive path. This behavior could be improved by combining it with directional antennas to guide movement more efficiently, the simulation uses isotropic antennas on all nodes to maintain consistency across scenarios. Additionally, if multiple attackers are present, the scenarios implement coordinated attacker strategies by sharing their estimated distanced so that they can comput trilateration via circle intersection to locate the transmitter node.

6.2 STATIC ATTACKER SCENARIOS

This section presents simulations where attackers remain fixed at their initial positions. Their only method of estimating the transmitter's position is through RSSI-based distance calculations. Since they lack directional information, they can only infer a circle of possible transmitter locations. These scenarios serve as baseline comparison before introducing the more advanced mobile attacker scenarios. Deception mechanisms will be introduced for every scenario type to evaluate their impact on distance estimatino.

6.2.1 SINGLE STATIC ATTACKER

Since this if the first attacker scenario, it serves as the simplest attacker model and establishes a baseline for comparison. More attackers and mobility will be added in later subsections, allowing for better accuracy in transmitter localization.

This scenario consists of a single passive attacker listening to the network at a fixed position. The attacker intercepts packets, measures their RSSI, and estimated the distance to the transmitter using the configured propagation loss model.

Distance estimation using RSSI and Path Loss Model

To compute the distance from the RSSI, the Free-Space Path Loss (FSPL) equation, which derives from Friis transmission formula, is applied:

$$P_r^{[dB]}(d) = P_t^{[dB]} + G_t^{[dBi]} + G_r^{[dBi]} + 20\ log_{10}\left(\frac{\lambda}{4\pi d}\right)$$

where:

- $P_r(d)$ is the received power at distance d (in dBm)
- P_t is the transmitted power (in dBm)
- G_t is the transmitter antenna gain (in dBi)
- G_r is the receiver antenna gain (in dBi)
- λ is the wavelength of the transmitted signal (in meters)
- d is the distance between the transmitter and receiver (in meters)

Since path loss is defined as the difference between transmitted power and received power:

$$PL\ (d) = P_t - P_r(d)$$

Substituting Friis' equation:

$$PL\ (d) = -\left(G_t + G_r + 20\ log_{10}\left(\frac{\lambda}{4\pi d}\right)\right)$$

Since frequency (f) and wavelength (λ) are related by the speed of light (c), and under standard conditions G_t and G_r are assumed to be 0 dBi, this simplifies to:

$$PL\,(d) = 20\,log_{10}\left(\frac{4\pi df}{c}\right) = 20\,log_{10}\left(\frac{4\pi f}{c}\right) + 20\,log_{10}(d)$$

At a typical reference distance of d_0 = 1m, the formula simplifies to:

$$PL_0 = PL\,(d = 1m) = 20\,log_{10}\left(\frac{4\pi f}{c}\right)$$

Thus, the general FSPL model becomes:

$$PL\,(d) = PL_0 + 20\,log_{10}\left(\frac{d}{d_0}\right)$$

The Friis' transmission formula models signal propagation in an unobstructed environment, free-space conditions assume that path loss follows an inverse-square law, meaning a path loss exponent of 2. In real-world environments, signal propagation is affected by obstacles, reflections and multipath effects, which causes the received signal to decay faster than in free space. To model this behavior, we introduce the path loss exponent n:

$$PL\,(d) = PL_0 + 10n\,log_{10}\left(\frac{d}{d_0}\right)$$

where n accounts for environmental attenuation:

- Free space: n = 2 (ideal conditions)
- Suburban environments: n = 3.0 (used in this simulation)
- Dense urban environments: n = 3.5 – 4.5
- Indoor propagation: n = 4 - 6

Solving for d, the equation used in the simulation to estimate the transmitter's distance is:

$$d = 10^{\frac{PL(d)-PL_0}{10n}} = 10^{\frac{P_t-P_r-PL_0}{10n}}$$

Simulation Parameters and Expected Behavior

With the following predefined values, the only remaining variable is the RSSI (P_r) or received power, which changes dynamically:

- Transmit power (P_t): 0 dBm
- Path loss exponent (n): 3.0
- Reference path loss at 1m (PL_0): - 46.6777 dBm[1]

Since the attacker is static, the estimated distance alone does not reveal the exact position of the transmitter, it only defines a circle of possible locations. Without deception, the RSSI-based calculation should exactly give the real distance. However, when deception mechanisms are introduced and network nodes apply power fluctuations in their transmissions, the attacker will miscalculate the distance. The degree of error in the intersection points increases as the difference between the network's predefined transmission power (0 dBm) and the fluctuated power (down to a maximum of -15 dBm) becomes larger. However, it is important to note that the transmission power cannot be reduced excessively. In real-world scenarios, hardware limitations and regulatory constraints can restrict the minimum power levels. Furthermore, reducing transmission power too much could cause legitimate network receivers, such as sensor nodes and the sink node, to fail to receive packets due to their sensitivity thresholds.

Figure 8 shows an example of a randomly generated network topology for this scenario, the initial node placement occurs in a 20x20 meter grid of the Single Static Attacker scenario. The sink node, represented in blue, act as the PAN coordinator and central receiver for sensor data collection, while the green node represents the sensor transmitting 50-byte packets every second. The red node indicates the static attacker, positioned to intercept wireless communications and to estimate distances based on RSSI values of the packets.

[1] These simulation parameters will be used across all simulations and will no change, apart from the transmission power fluctuations present in deception scenarios

The grid layout demonstrates a typical short-range IEEE 802.15.4 deployment, aligning with real-world applications of low-power WSNs. The attacker node, located at a fixed position, provides a circular estimation of the transmitter's location using the Free-Space Path Loss (FSPL) model.

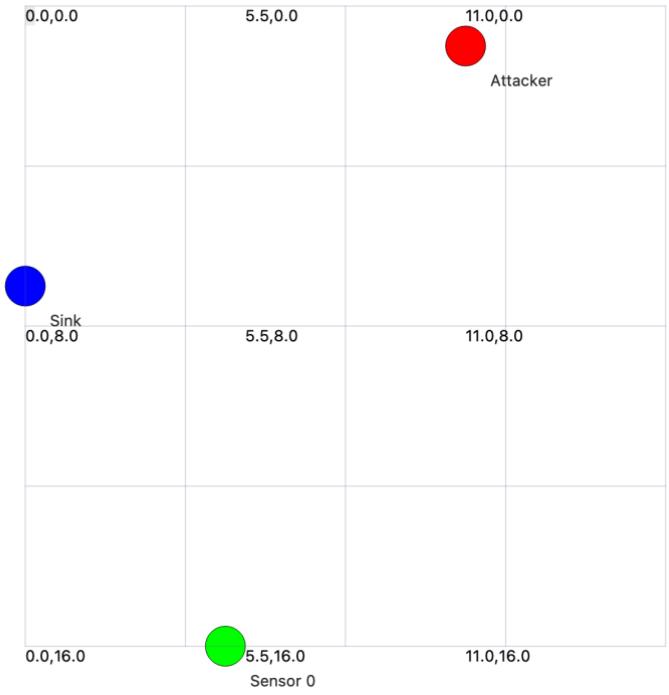


Figure 8. NetAnim example of the network topology in the Single Static Attacker scenario

The main limitation of a single attacker is that it can only provide with a circular estimation of the transmitter position, adding a second attacker enables circle intersection, reducing localization uncertainty.

6.2.2 TWO STATIC ATTACKERS

This scenario introduces a second passive attacker positioned statically, both are listening to the network traffic. The presence of two attackers allows for significant improvements in

transmitter localization since they can combine their distance estimations to compute an intersection region with two possible points for the transmitter position.

Distance Estimation and Circle Intersection

As in the previous scenario, both attackers individually estimate their distances to the transmitter using the RSSI values and the Free-Space Path Loss model. However, with two attackers, the process is more complex since it requires finding the intersection of the two circles formed by the attackers' distance estimates. The calculation steps are as follows:

1. **Circle definitions**

- Attacker 1: center = (x_0, y_0), radius = r_0
- Attacker 2: center = (x_1, y_1), radius = r_1
- Unknown: points of intersection A and B

2. **Distance between circle centers**

The distance between the two circles' centers is computed as:

$$d = \sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2}$$

3. **Intersection possibilities**

The relationship between d, r_0 and r_1 determines whether the circles intersect:

- No solution if $d > r_0 + r_1$ (circles are separate)
- No solution if $d < |r_0 - r_1|$ (one circle is contained within the other)
- One solution if $d = r_0 + r_1$ or $d = |r_0 - r_1|$ (circles touch)
- Two solutions if $|r_0 - r_1| < d < r_0 + r_1$ (circles intersect at two points)

4. **Finding point P**

To compute the intersection points, we first find the point P = (P_x, P_y) on the line connecting the two circle centers, at a distance **a** from the center of Circle 1:

$$(I)\ r_0^2 = a^2 + h^2$$

$$(II)\ r_1^2 = (d-a)^2 + h^2$$

Solving for **a**, we obtain:

$$a = \frac{r_0^2 - r_1^2 + d^2}{2d}$$

We now compute the point P, which lies on the line connecting the centers of the two circles. This point represents the location where the line of the two centers intersects the line that connects the two potential intersection points of the circles. To calculate P, we start at the center of Circle 1 and move along the direction of the line connecting the two circle centers. This direction is represented as a unit vector $\frac{x_1-x_0}{d}$, scaled by the distance **a** from the center of Circle 1 to P. For a better understanding of this process, refer to Figure 9

$$P = (P_x, P_y) = \left(x_0 + a\frac{x_1 - x_0}{d}, y_0 + a\frac{y_1 - y_0}{d}\right)$$

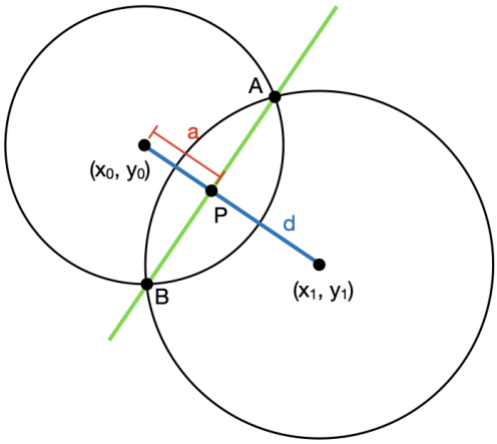


Figure 9. Intersection of two attackers' estimated distances to determine potential transmitter locations A and B

5. Compute distance from P to intersection points

The perpendicular distance from P to the intersection points (length of segments $\overline{PA}$ or $\overline{PB}$) is:

$$h = \sqrt{r_0^2 - a^2}$$

6. Finding intersection points

To calculate the offsets (r_x, ry), which represent the perpendicular distance from point P to the intersection points, a unit vector perpendicular to the line between the centers is scaled by **h**, the height. These offsets are added and subtracted from P to find points A and B.

$$r_x = -h\,\frac{y_1 - y_0}{d}, \qquad r_y = h\,\frac{x_1 - x_0}{d},$$

where **d** is the distance between the circle centers. These offsets are then added and subtracted from P to determine the two intersection points.

$$A = (P_0 + r_x,\ P_1 + r_y)$$

$$B = (P_0 - r_x,\ P_1 - r_y)$$

This process allows the attackers to estimate two potential locations for the transmitter, reducing the uncertainty significantly compared to the single-attacker scenario.

Expected Behavior

In this scenario, the two attackers are statically positioned within the 20x20 meter grid. Each attacker intercepts packets from the network calculates its distance to the packet transmitter node using the FSPL model. Without any deception mechanisms, the two circles created from their distance estimates should intersect at two points: one representing the transmitter's actual location and the other an incorrect position due to the mathematical ambiguity of the circle intersection.

However, when deception mechanisms are introduced, the nodes in the network intentionally fluctuate their transmission power. This fluctuation distorts the RSSI values received by the attackers, resulting in incorrect distance calculations. Consequently, the intersection points of the two circles, A and B, will deviate from the actual transmitter's location. Excessive reduction of the transmission power may prevent legitimate network nodes from receiving packets, as their RSSI values would fall below the minimum sensitivity threshold.

Figure 10 demonstrates a randomly generated network topology in the Two Static Attackers scenario. The green sensor node continuously transmits packets to the blue sink node. The two red nodes represent the static attackers, positioned to intercept packets and estimate the transmitter's location using RSSI values and then, compute the intersection.

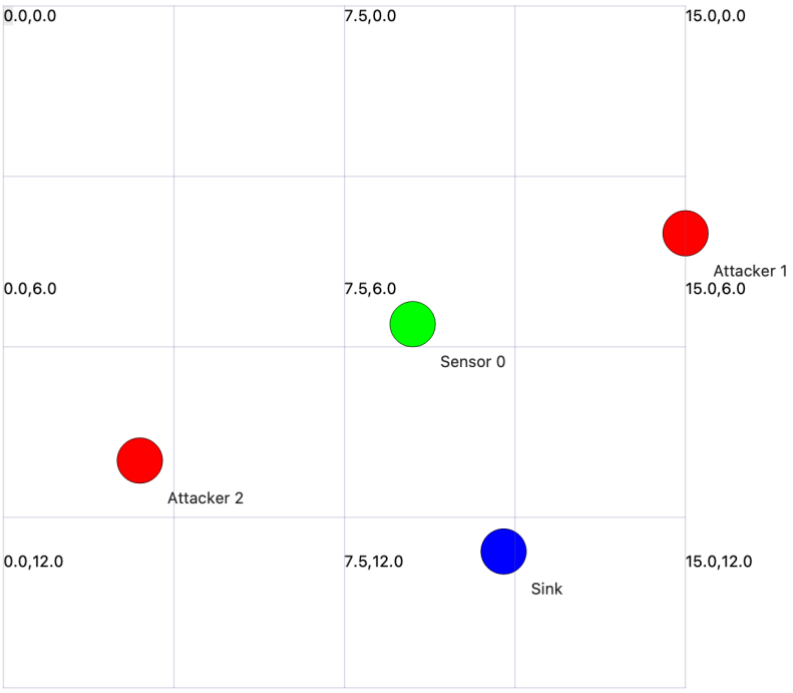


Figure 10. NetAnim example of the network topology in the Double Static Attacker scenario

The addition of a second attacker demonstrates the potential of coordinated distance estimation to significantly reduce localization uncertainty. However, the two-point ambiguity remains a limitation. The introduction of a third attacker in the next scenario addresses this limitation by enabling unique localization through trilateration.

6.2.3 THREE STATIC ATTACKERS

This scenario introduces a third passive static attacker to the network. It is important to note that the three attackers must not be positioned in a straight line for trilateration to succeed. Trilateration is a geometric method commonly used in wireless networks to determine the position of a node based on distances from three known reference points. The addition of a third attacker allows for this unique localization of the transmitter, as the intersection of three distance estimates, i.e., three circles, results in a single solution. Trilateration eliminates the ambiguity present in the previous scenario where two attackers produced two possible solutions for the transmitter locations.

Distance Estimation and Trilateration

With three attackers, the distance estimates define three circles, and the transmitter's position corresponds to the single intersection point of these circles. The mathematical process to compute this unique solution is as follows:

1. **Circle definitions**

- Attacker 1: center = (x_1, y_1), radius = r_1
- Attacker 2: center = (x_2, y_2), radius = r_2
- Attacker 3: center = (x_3, y_3), radius = r_3
- Unknown: 1 solution = (x, y)

2. **Distance equations**

The basic distance equations for the three attackers are based on the Euclidean distance formula:

$$(I)\ (x - x_1)^2 + (y - y_1)^2 = r_1^2$$

$$(II)\ (x - x_2)^2 + (y - y_2)^2 = r_2^2$$

$$(III)\ (x - x_3)^2 + (y - y_3)^2 = r_3^2$$

3. Deriving trilateration formulas

Eliminate the quadratic terms by subtracting the equations from each other:

- Susbtract Equation I from Equation II

$$(IV)\ 2x(x_1 - x_2) + 2y(y_1 - y_2) = r_2^2 - r_1^2 - (x_2^2 - x_1^2 + y_2^2 - y_1^2)\ ;$$

This equation will be represented as:

$$Ax + By = C$$

- Susbtract Equation II from Equation III

$$(V)\ 2x(x_2 - x_3) + 2y(y_2 - y_3) = r_3^2 - r_2^2 - (x_3^2 - x_2^2 + y_3^2 - y_2^2);$$

This equation will be represented as:

$$Dx + Ey = F$$

4. Solving the System of Equations

The two linear equations ($Ax + By = C$ and $Dx + Ey = F$) are solved using Cramer's Rule to find the unique intersection point (x, y).

First, the determinant is calculated as:

$$Det = AE - BD$$

If Det is zero or very small (indicating a singular matrix or parallel lines), trilateration cannot proceed. Then, solve for x and y using Cramer's Rule to obtain the unique intersection point of the three circles:

$$x = \frac{\begin{vmatrix} C & B \\ F & E \end{vmatrix}}{Det}; \quad y = \frac{\begin{vmatrix} A & C \\ D & F \end{vmatrix}}{Det}$$

This process determines the transmitter's exact location when all three circles intersect at a single point. Figure 11 illustrates the trilateration process.

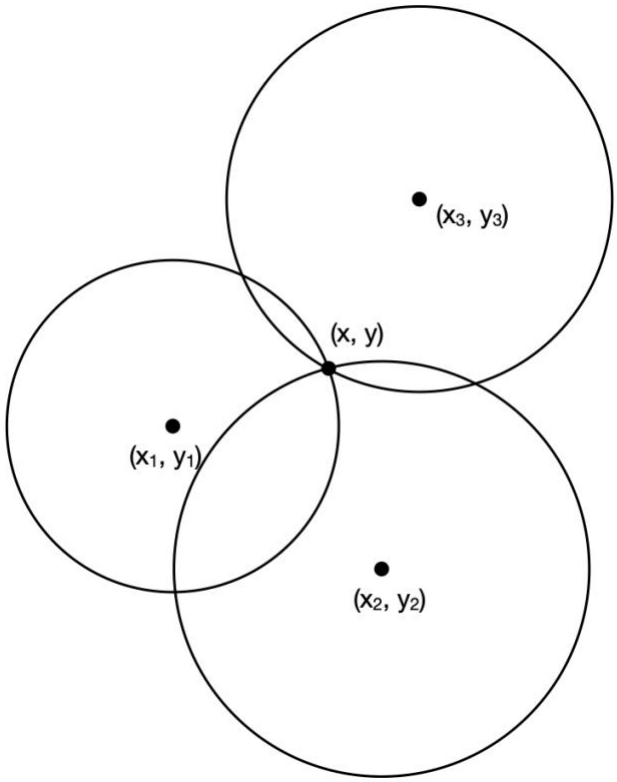


Figure 11. Trilateration with three attackers estimating the transmitter's location through the intersection of three circles

Expected Behaviour

In this scenario, the three attackers are statically positioned within the 20x20 meter grid. Each attacker intercepts packets and estimates its distance to the transmitter. Without deception, the three circles defined by the attackers' distance estimates intersect uniquely at the transmitter's location, enabling precise localization.

When deception mechanisms are introduced, the fluctuating transmission power of the network nodes distorts the RSSI values received by the attackers. This results in incorrect distance estimates, causing the trilateration process to produce a location that deviates from the actual transmitter position. The error in transmitter localization grows as the transmission power fluctuates more significantly from the baseline 0 dBm, with a maximum reduction to -15 dBm. However, in real-world deployments, excessive power reduction is constrained by hardware limitations and receiver sensitivity thresholds, as signals below the sensitivity level could become undetectable.

Figure 12 presents an example of a randomly generated network topology from the Monte Carlo simulations. The green node represents the sensor transmitting data to the blue sink node, while the three red nodes are the static attackers positioned to intercept packets. These attackers collaborate to compute the transmitter's exact location using trilateration.

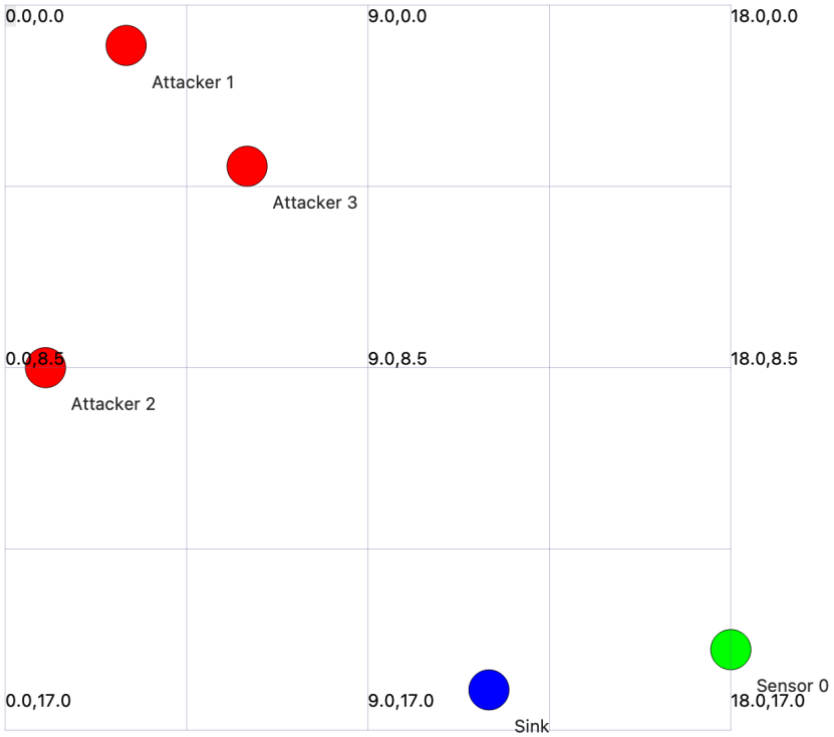


Figure 12. NetAnim example of the network topology in the Triple Static Attacker scenario

6.3 MOBILE ATTACKER SCENARIOS

In contrast to the previous static attacker scenarios, this section presents cases in which attackers are mobile and dynamically attempt to track the transmitter's position. Unlike static attackers, which rely on RSSI-based distance estimation from a fixed position, mobile attackers base their decisions on real-time variations of the signal strengh as they move.

The method used to compute the estimated position of the transmitter remains the same as in the previous section with static attackers. A single mobile attacker esimated its distance based on RSSI but, without knowing the direction of the transmitter incoming signal, it must rely on movement decisions based on RSSI trends. When two attackers are present, they use circle intersection, similar to the static two-attacker scenario, to determine two possible locations for the transmitter. Finally, three mobile attackers leverage trilateration to compute a unique intersection point, allowing for a more precise estimation. The key difference in these mobile scenarios is that, unlike static attackers who work with a single snapshot of RSSI, mobile attackers continuously update their estimates over time, refining their position as they move.

The mobility model implemented in these simulations is reactive, meaning that attackers do not follow predetermined paths but rather adapt their trajectory in response to observed RSSI values. As in the previous scenarios, deception mechanisms will be introduced to analyze how transmission power fluctuations influence an attacker's ability to track the transmitter accurately. The following subsections explore the behavior of single, two, and three mobile attackers.

As in the static attacker scenarios, the network topology is randomly generated for each simulation run following the Monte Carlo methodology. This ensures a diverse set of initial node placements, allowing for a robust performance evaluation. Since the focus of this section is on the system's mobility logic rather than specific instance visualizations, no topology figures will be included here.

6.3.1 SINGLE MOBILE ATTACKER

This scenario introduces a single mobile attacker that continuously listens to network traffic and moves based on RSSI variations. Unlike static attackers, who could only estimate distance but not direction, the mobile attacker assumes that an increasing RSSI value means it is moving in the correct direction, while a decreasing or equal RSSI value indicates it should adjust its trajectory. The attacker begins at an initial random position and makes movement decisions as follows.

At the start of the simulation, the attacker receives packets and estimates its distance to the transmitter. Since it lacks a predefined path, its initial movement direction is chosen randomly. After each RSSI measurement, the attacker evaluates whether it is getting closer to or farther from the transmitter. If the RSSI increases, it continues in the same direction. If the RSSI decreases, it assumes it is moving away from the target and randomly selects a new direction.

The attacker's movement speed is adaptive, meaning that when it is far from the transmitter, it moves relatively fast, but as it gets closer, it slows down. This behavior mimics real-world tracking strategies, where an adversary refines its movements as it gets closer to the target and prevents excessive oscillations near the estimated location. To prevent the attacker from traveling too far outside the simulation area, movement is restricted within a 500m x 500m boundary. If the attacker reaches the boundary, it reverses direction.

These boundary mechanism as well as the controlled approach towards the estimation distance or position are present in all scenarios for mobile attackers.

Without deception, the attacker is expected to gradually approach the transmitter, refining its distance estimates over time. However, when deception mechanisms are introduced, the attacker will experience misleading RSSI values due to transmission power fluctuations. This misleads the attacker into making incorrect movement decisions, either overestimating or underestimating its distance to the transmitter. Since the attacker has only a single distance

estimate at any given time, it cannot determine precise coordinates, but it can attempt to minimize uncertainty by refining its movement.

6.3.2 TWO MOBILE ATTACKERS

This scenario introduces a second mobile attacker, allowing for a more sophisticated tracking approach through collaborative distance estimation. Each attacker individually estimates its distance to the transmitter using the same RSSI-based methodology as the single mobile attacker. However, when two attackers are present, they can share their estimated distances and compute circle intersections to further refine their movement.

The two attackers collaborate by computing circle intersections of their estimated distance circles. Each attacker measures its RSSI-based distance and, upon receiving the same packet, they identify a common transmitter. This is made possible by the transmitter ID tagging mechanism introduced in the system overview, ensuring that both attackers are computing distances to the same node. In a real-world scenario, this tagging mechanism would translate to both attackers passively sniffing network traffic simultaneously after identifying the transmission channel, ensuring that they capture the same packets for distance estimation. Using their distance estimates, they compute the intersection of their respective circles, resulting in two possible locations for the transmitter.

To resolve the ambiguity between the two intersection points, each attacker moves toward a different candidate location in a controlled manner. This movement is constrained by real-world considerations; attackers cannot instantly reach a new position between two packet receptions, as in real deployments there is a delay of milliseconds or a few seconds between packet transmissions. This gradual movement helps the attackers refine their localization by continuously updating their distance estimations.

As in previous scenarios, deception mechanisms introduce power fluctuations, leading to incorrect distance estimates. This causes the intersection points to shift over time, making localization more challenging for the attackers.

6.3.3 THREE MOBILE ATTACKERS

The final scenario expands the approach to three mobile attackers, enabling real-time trilateration for continuous tracking of the transmitter's position. The addition of a third attacker allows for a single unique intersection, removing ambiguity.

Each attacker estimates its distance to the transmitter based on RSSI and follows the same reactive decision-making model as described in previous scenarios. However, instead of independently moving based on RSSI trends alone, the three attackers coordinate by computing trilateration after receiving the same packet. Once all three attackers have obtained an RSSI measurement from the same transmitter, they exchange their estimated distances and determine the unique location where their three distance estimates intersect.

After computing the intersection, each attacker moves toward this estimated position in a controlled manner. Specifically, each attacker follows the straight-line path between its current position and the computed intersection point, advancing 50% of the distance to the target in that iteration. To introduce variability in movement, a random factor between 0.8 and 1.2 is applied to this displacement, ensuring that the attackers' movement remains dynamic rather than entirely deterministic.

Initially, one might assume that introducing a third mobile attacker would lead to greater resilience against deceptive power fluctuations. The redundancy of three independent RSSI measurements should, in theory, help mitigate errors introduced by fluctuating transmission power. However, if a node transmits a packet with an RSSI value several dBm lower than expected, all three attackers will overestimate their distances. This causes their computed circles to expand significantly, resulting in an estimated transmitter location that is even further from the actual position. In such cases, rather than improving accuracy, the coordination may contribute to greater localization errors, thus, more complex attacker behaviour must be proposed to deal with the power fluctuation defensive mechanism.

Additionally, to prevent attackers from drifting indefinitely due to compounding estimation errors, a 1000x1000 meter boundary has been established within the simulation. Without

this constraint, attackers could move arbitrarily far away from the transmitter as they repeatedly recompute erroneous distances. The controlled movement approach, which limits attackers to advancing 50% of the estimated distance toward the computed intersection, further ensures that tracking remains within realistic limits and prevents abrupt and long-step movements. This constraint is necessary to prevent attackers from leaving the search area entirely and will be further analyzed Analysis of Results section.

7. ANALYSIS OF RESULTS

7.1 OVERVIEW

This section provides a comprehensive analysis of the simulation results across all scenarios to evaluate the effectiveness of the deception mechanism and its impact on localization accuracy. The key objectives of this analysis are the following:

- To compare localization accuracy with and without deception mechanisms.
- To assess the influence of increasing the number of attackers, both static and mobile, on localization performance.
- To examine the role of mobility and coordination among attackers in improving or hindering their ability to estimate transmitter positions.

The metrics used to evaluate these scenarios include Mean Absolute Error (MAE) and Root Mean Square Error (RMSE), which quantify the deviation between the estimated and actual positions or distances to the transmitter. These metrics provide a detailed view of the accuracy of attacker localization under varying conditions. In addition, the error distribution across all simulation runs was analyzed to identify patterns and outliers, providing deeper insights into the scenarios' robustness.

For every scenario, histograms of absolute error and the distribution of MAE values across 1,000 Monte Carlo simulations were generated. The MAE captures the average error magnitude, while the RMSE emphasizes larger deviations by assigning greater weight to outliers.

7.2 RESULTS BY SCENARIO

Each scenario consists of 1,000 simulations conducted over a 30-second duration, with and without deception. The use of Monte Carlo methods ensures a robust representation of

metrics with the goal of providing comprehensive insights into localization accuracy and the impact of deception mechanisms.

7.2.1 SINGLE STATIC ATTACKER

This scenario involves a single static attacker attempting to localize the transmitter based on RSSI measurements. The primary objective is to evaluate the impact of deception mechanisms on the accuracy of the attacker's distance estimation.

Without Deception

All estimated distances across the 1000 simulations without deception have an error close to zero. This is caused by the accurate knowledge of transmission parameters, which ensures precise distance estimation. The results confirm the attacker's ability to localize the transmitter with minimal deviations in ideal, deception-free conditions.

With Deception

The table below summarizes the key metrics obtained for this scenario:

Table 3. Key metrics for single static attacker scenario with deception

Metric	Value
Mean Absolute Error (MAE)	12.82m
Root Mean Square Error (RMSE)	15.53m
Standard Deviation Error	8.77
Minimum error	-52.07
25%	-16.95
50%	-10.36

75%	-6.36
Maximum error	-0.47

The metrics in Table 3 reveal that deception mechanisms had a noticeable impact on the attacker's localization accuracy:

- The Mean Absolute Error (MAE) of 12.82 meters indicates that the average error in distance estimation is relatively high for a small-scale network.
- The Root Mean Square Error (RMSE) of 15.53 meters, which penalizes large errors, suggests the presence of significant deviations in certain cases.
- The error distribution percentiles (25%, 50%, 75%) show that a majority of errors lie between -16.95 and -6.36 meters, with extreme values like -52.07 meters and -0.47 meters representing outliers.

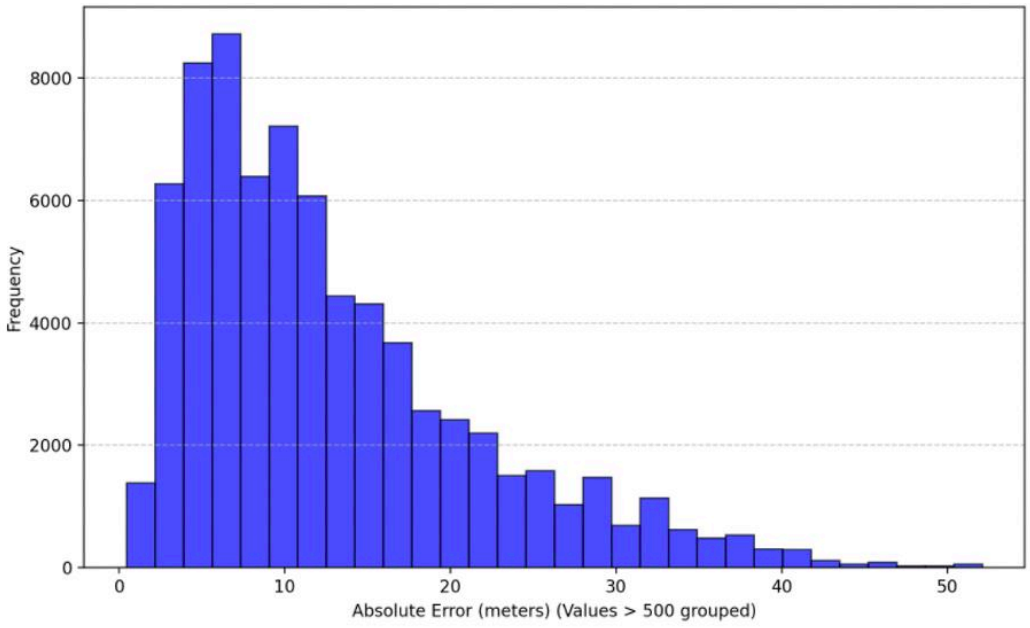


Figure 13. Single static attacker with deception - Histogram of clipped absolute errors across all simulations

Figure 13 provides a detailed view of the error distribution across all simulations. The histogram shows the following key patterns:

- Most errors are concentrated below 20m, with a gradual tapering as the error increases.
- There are a few outliers with absolute errors reaching up to 50 meters, highlighting instances where the deception mechanism was highly effective
- The right-skewed distribution demonstrates that while the attacker often achieves moderate accuracy, the deception mechanism introduces a wide range of potential errors.

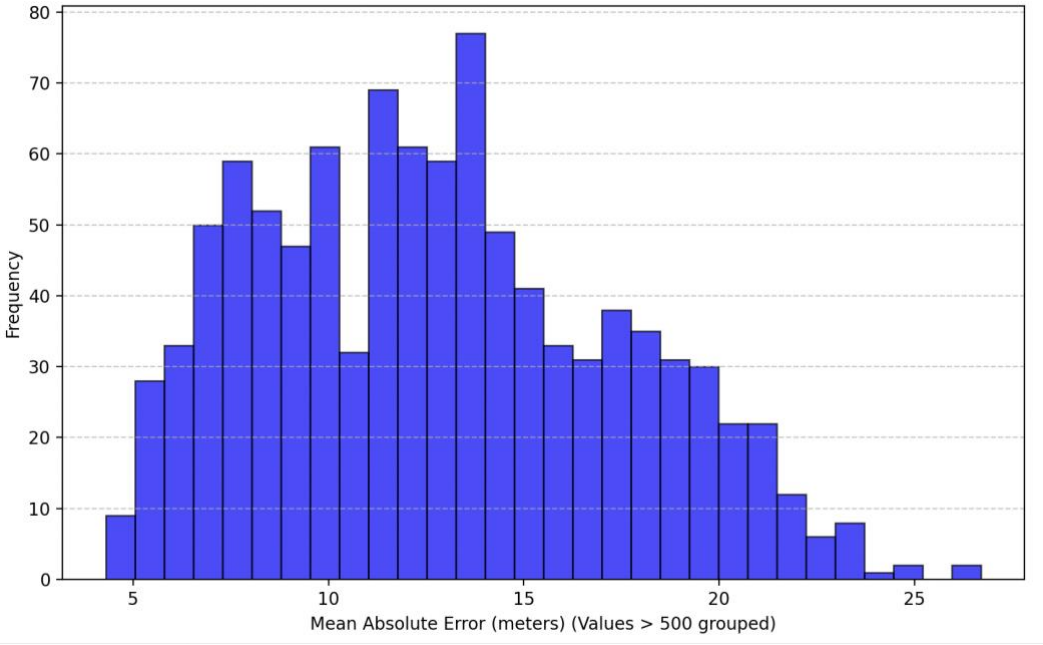


Figure 14. Single static attacker with deception - Histogram of Clipped MAE Across Simulations

Figure 14 illustrates the variation in Mean Absolute Error across individual simulations. Most simulations exhibit a MAE between 5 and 20 meters, consistent with the overall mean. Another key insight is that a smaller number of simulations have MAE values exceeding 20 meters, suggesting that the random initial positioning and power fluctuation patterns significantly influence attacker performance.

This histogram underscores the impact of simulation variability on performance metrics, emphasizing the importance of Monte Carlo methods in evaluating the system's robustness.

The results from this scenario highlight the effectiveness of the deception mechanism in degrading localization accuracy for a single static attacker. The consistent negative mean error of -12.82 meters reflects the intentional design of the power fluctuations, which reduce transmission power below the default level, leading the attacker to consistently overestimate the distance to the transmitter. The significant spread of errors, as evidenced by the standard deviation of 8.77 meters, underscores the variability introduced by these fluctuations, which creates substantial uncertainty in distance estimations.

7.2.2 TWO STATIC ATTACKER

This scenario involves two static attackers attempting to localize the transmitter based on RSSI measurements. The primary objective is to evaluate how adding a second attacker improves localization accuracy and to examine the impact of deception mechanisms on this improvement.

Without deception

In the absence of deception mechanisms, the attackers benefit from accurate RSSI values that allow for precise distance estimations. The circle intersection approach narrows down the transmitter's location to two possible points: one very close to the true position and the other farther away. Since the error has been calculated as the mean of the errors for the two potential positions, the resulting error metrics reflect this averaging, where one error value is close to zero due to the correct solution.

Table 4. Key metrics for two static attackers scenario without deception

Metric	Value
Mean Absolute Error (MAE)	6m
Root Mean Square Error (RMSE)	7.34m

Standard Deviation Error	4.23m
Minimum error	0.00
25%	2.54m
50%	5.34m
75%	8.5m
Maximum error	22.91m

The results in Table 4 demonstrate the high accuracy achievable with two static attackers when no deception mechanisms are in place:

- Mean Absolute Error (MAE): the MAE of 6.00 meters represents the average distance estimation error across all simulations. This is a significant improvement compared to the single attacker scenario because the addition of a second attacker reduces localization uncertainty by leveraging circle intersections.
- Root Mean Square Error (RMSE): at 7.34 meters, the RMSE highlights the relatively low impact of larger deviations in error values.
- Error distribution: the standard deviation of 4.23 meters indicates some variability in the error values, which can be attributed to the random network topologies generation.
- Percentile distribution: the 25th, 50th, and 75th percentiles show that most errors fall within 2.54 meters to 8.50 meters, with a median error of 5.34 meters. The minimum error is close to zero, reflecting the consistent identification of the correct transmitter location in each simulation.

With Deception

The table below summarizes the key metrics obtained from the simulation:

Table 5. Key metrics for two static attackers scenario with deception

Metric	Value
Mean Absolute Error (MAE)	19.41m
Root Mean Square Error (RMSE)	22.04m
Standard Deviation Error	10.44m
Minimum error	2.31m
25%	11.63m
50%	17.21m
75%	24.68m
Maximum error	69.15m

Table 5's results demonstrate the significant impact of deception mechanisms on localization accuracy:

- High MAE and RMSE: the Mean Absolute Error (19.41 meters) and Root Mean Square Error (22.04 meters) indicate a considerable degradation in localization accuracy compared to scenarios without deception. The RMSE highlights the presence of large deviations caused by power fluctuations.
- Wide error spread: the standard deviation (10.44 meters) reflects a high variability in errors, likely due to dynamic transmission power changes affecting the attackers' RSSI measurements affecting the compound error of both attackers.

- Error range: the error values range from as low as 2.31 meters to as high as 69.15 meters, indicating that while the deception mechanism was not uniformly effective across all packets, it introduced enough noise to create substantial uncertainty.

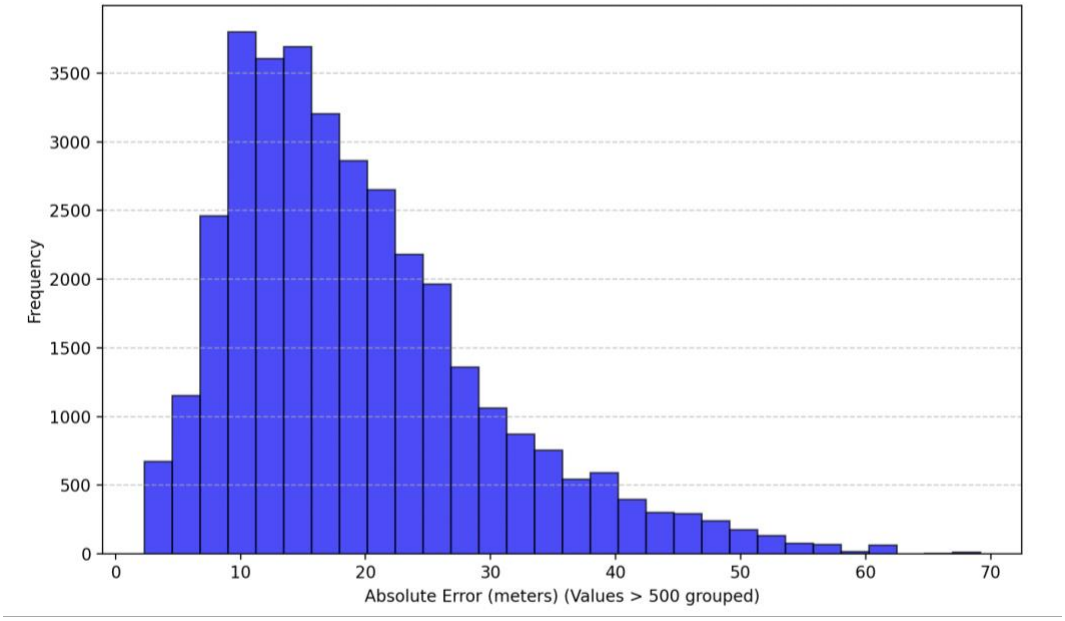


Figure 15. Two static attackers with deception - Histogram of clipped absolute errors across all simulations

Figure 15 shows that most errors are concentrated between 8 and 30 meters, with a peak around 15 meters, which demonstrates that the deception mechanism frequently causes moderate errors in localization. Additionally, the tail of the distribution extends to around 60 meters, reflecting instances where the power fluctuations resulted in severe inaccuracies.

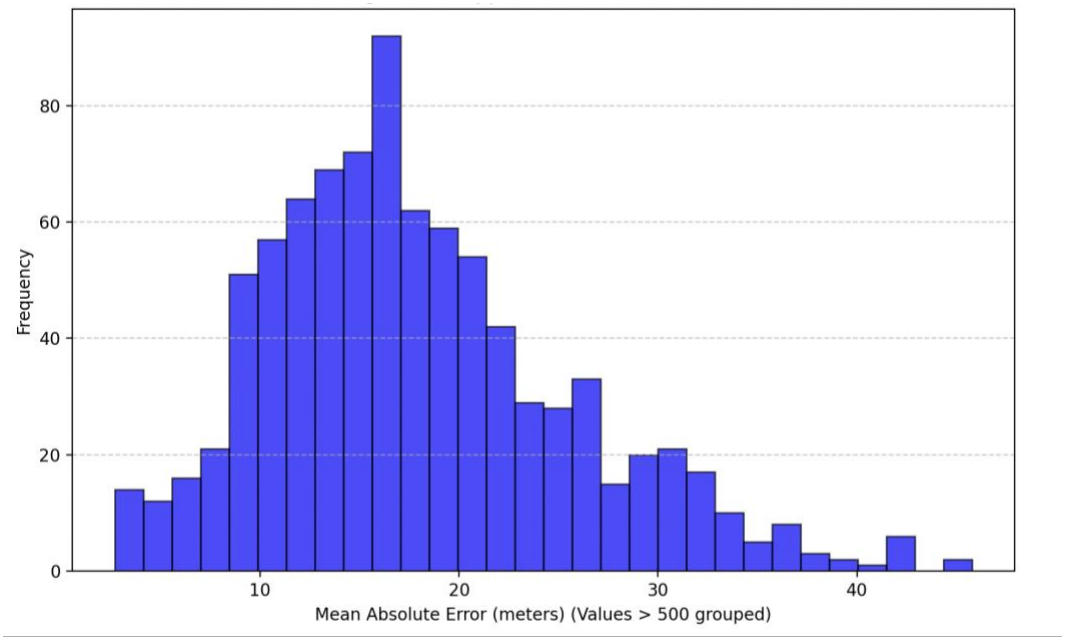


Figure 16. Two static attackers with deception - Histogram of Clipped MAE Across Simulations

Figure 16, the histogram of clipped MAE across simulations, highlights the variability in performance across individual simulations. Most simulations have a MAE between 8 and 25 meters, consistent with the overall mean of 19.41 meters. A smaller number of simulations exhibit an MAE exceeding 30 meters, suggesting that specific random configurations of initial positions and power fluctuations had a disproportionately large impact on the attackers' accuracy.

This scenario illustrates that while two coordinated attackers are better equipped to localize the transmitter compared to a single attacker, the effectiveness of their coordination is significantly reduced by deception mechanisms. By fluctuating the transmission power, the network nodes disrupt the attackers' RSSI-based distance estimations, resulting in higher variability and larger errors. The coordination strategy, which resolves ambiguity by computing the intersection of distance circles, is particularly susceptible to error amplification when both attackers are misled by deceptive fluctuations.

7.2.3 THREE STATIC ATTACKER

In this scenario, three static attackers collaborate to localize the transmitter using trilateration. With three distance estimates, the attackers can compute a unique intersection point, eliminating the ambiguity observed in the two-static-attacker case.

Without Deception

Without deception, the trilateration method achieves perfect localization accuracy. All error metrics, including the Mean Absolute Error (MAE) and Root Mean Square Error (RMSE), are effectively zero. This confirms that in ideal conditions, with undistorted RSSI values, the attackers can precisely determine the transmitter's location. This serves as the baseline for analyzing the effects of deception mechanisms in subsequent scenarios.

With Deception

Table 6 summarizes the key metrics for the scenario with deception:

Table 6. Key metrics for three static attackers scenario with deception

Metric	Value
Mean Absolute Error (MAE)	80.83m
Root Mean Square Error (RMSE)	265.75m
Standard Deviation Error	253.17m
Minimum error	0.21m
25%	16.80m
50%	35.16m
75%	70.07m

Maximum error	5904.35m

The metrics reveal the following key insights:

- The Mean Absolute Error (MAE) of 80.83 meters indicates a substantial average deviation in the attackers' distance estimates.
- The Root Mean Square Error (RMSE) of 265.75 meters highlights that extreme errors significantly impact the attackers' localization accuracy, as RMSE penalizes larger deviations.
- The high Standard Deviation (253.17 meters) underscores the variability in error, reflecting the impact of deception mechanisms on the attackers' trilateration process.
- The maximum error of 5904.35 meters suggests that in some simulations, the deception mechanism completely misled the attackers, resulting in extreme misestimation.

Additionally, the number of extreme errors (greater than 500 meters) is as follows:

- Total across all simulations: 1506 out of 74,175 values.
- Average per simulation: 19 out of 74,175 values.

These extreme errors reflect the ability of the deception mechanism to significantly mislead the attackers in certain scenarios.

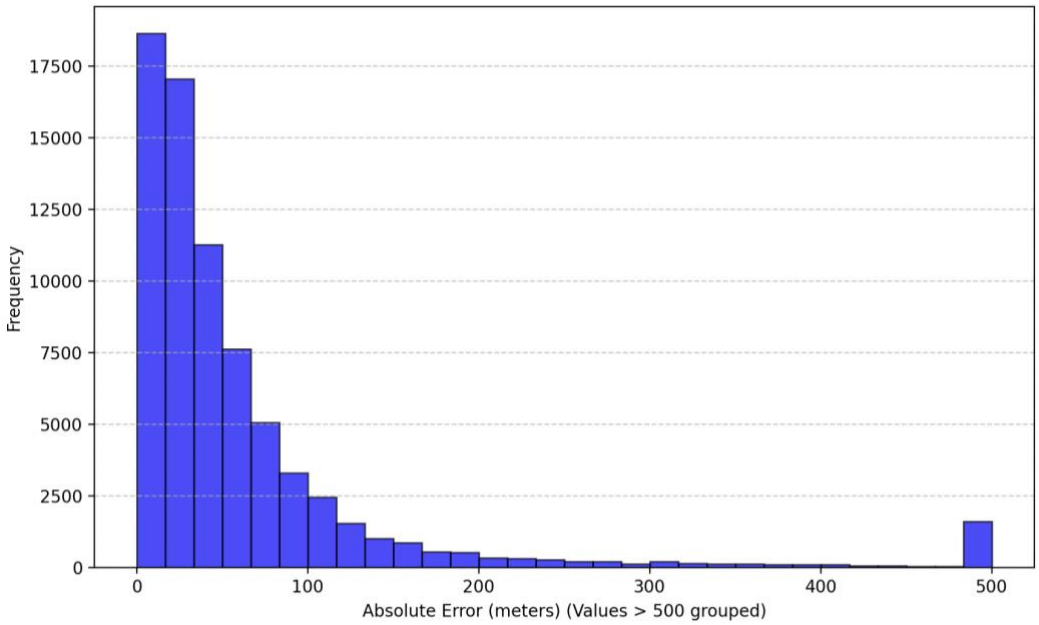


Figure 17. Three static attackers with deception - Histogram of clipped absolute errors across all simulations

Figure 17 visualizes the distribution of absolute errors across the simulations. The histogram reveals that most errors are concentrated below 100 meters, but a significant tail extends to much larger values, indicating that the deception mechanism misleads the attackers in some instances. The clustering of errors near the lower end suggests that while the attackers often achieve moderate accuracy, extreme errors occasionally occur.

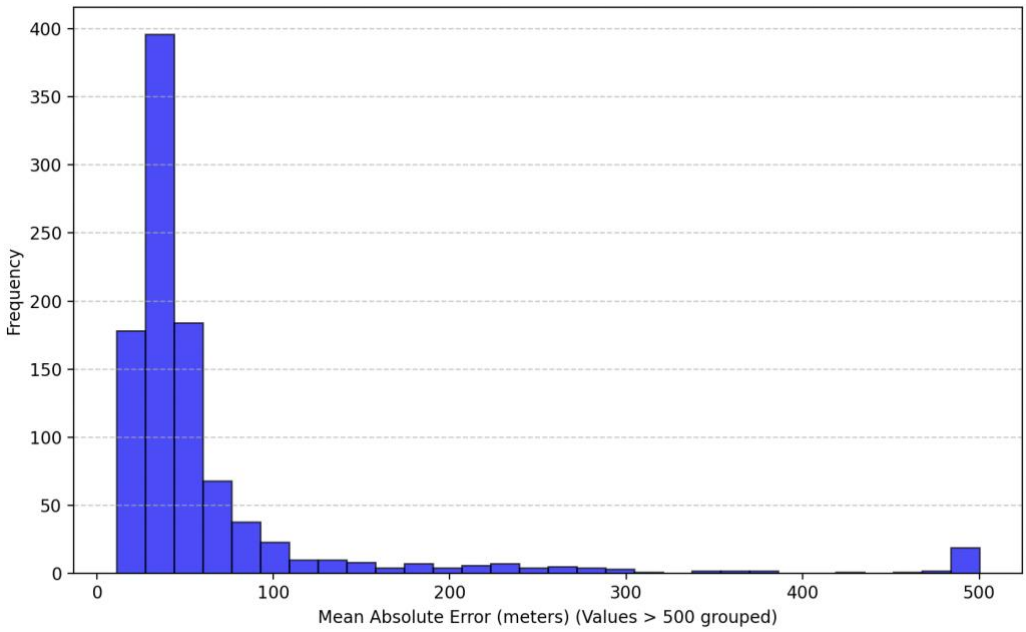


Figure 18. Three static attackers with deception - Histogram of Clipped MAE Across Simulations

The histogram in Figure 18 demonstrates that the majority of simulations exhibit MAE values below 100 meters, but a noticeable proportion of simulations have MAE values exceeding 200 meters, specially exceeding the clipping value of 500. This highlights the influence of both the network topology and the randomness of the deception mechanism on localization performance.

The high MAE and RMSE values, coupled with the significant number of extreme errors, reveal that the deception mechanism is highly effective in degrading the accuracy of coordinated trilateration by three static attackers. While trilateration theoretically offers precise localization by eliminating ambiguity from two-intersection scenarios, the fluctuating power levels result in overestimations of distance. This is reflected in the high standard deviation (253.17 meters) and the occasional extreme errors.

7.2.4 SINGLE MOBILE ATTACKER

The single mobile attacker scenario evaluates the ability of a single attacker to localize the transmitter while moving reactively based on RSSI trends. Unlike the static scenarios, this setup introduces mobility, enabling the attacker to refine its position over time.

Without Deception

In this scenario, the attacker achieves near-perfect localization accuracy, with practically all errors being close to zero. The Mean Absolute Error (MAE) is 0.01 meters, and the Root Mean Square Error (RMSE) is 0.05 meters, reflecting minimal deviations. The error distribution confirms that the majority of values are effectively zero, demonstrating the attacker's ability to consistently estimate distances accurately in the absence of deception mechanisms. Due to the negligible errors, no figures or histograms are included.

With Deception

Table 7 summarizes the key metrics for the scenario with deception:

Table 7. Key metrics for single mobile attacker scenario with deception

Metric	Value
Mean Absolute Error (MAE)	65.47m
Root Mean Square Error (RMSE)	94.31m
Standard Deviation Error	67.87m
Minimum error	-505.15m
25%	-91.28
50%	-39.93m
75%	-16.84m

Maximum error	-0.47m

- MAE of 65.47m demonstrates the significant challenge posed by deception, as the attacker consistently overestimates the distance to the transmitter.
- RMSE of 94.31m indicates the presence of substantial errors, with large deviations skewing the results in certain simulations.
- Standard deviation of 67.87m reflects the variability introduced by the randomized nature of deception, emphasizing inconsistent performance across simulations.
- Percentile Analysis: The median error of -39.93m and the 75th percentile error of -16.84m indicate that the majority of simulations produce notable overestimations of distance. However, the extreme minimum value of -505.15m shows cases where deception causes the attacker to make exceptionally inaccurate estimates.

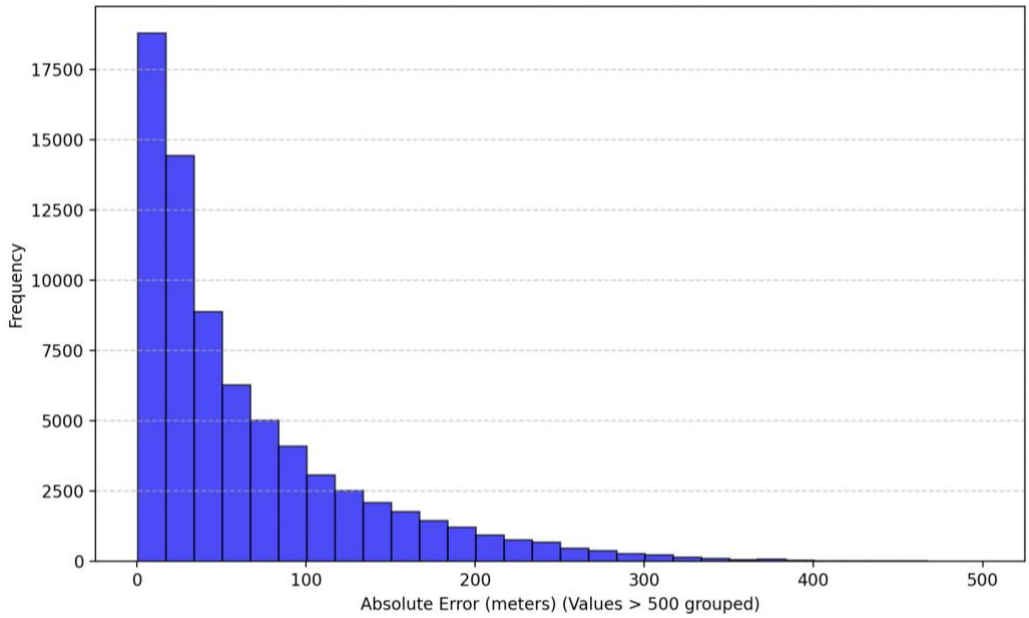


Figure 19. Single mobile attacker with deception - Histogram of clipped absolute errors across all simulations

Figure 19 depicts the distribution of absolute errors across all simulations. The concentration of errors below 200 meters demonstrates the general effectiveness of deception in creating moderate-to-severe localization inaccuracies, while a long tail highlights occasional extreme errors.

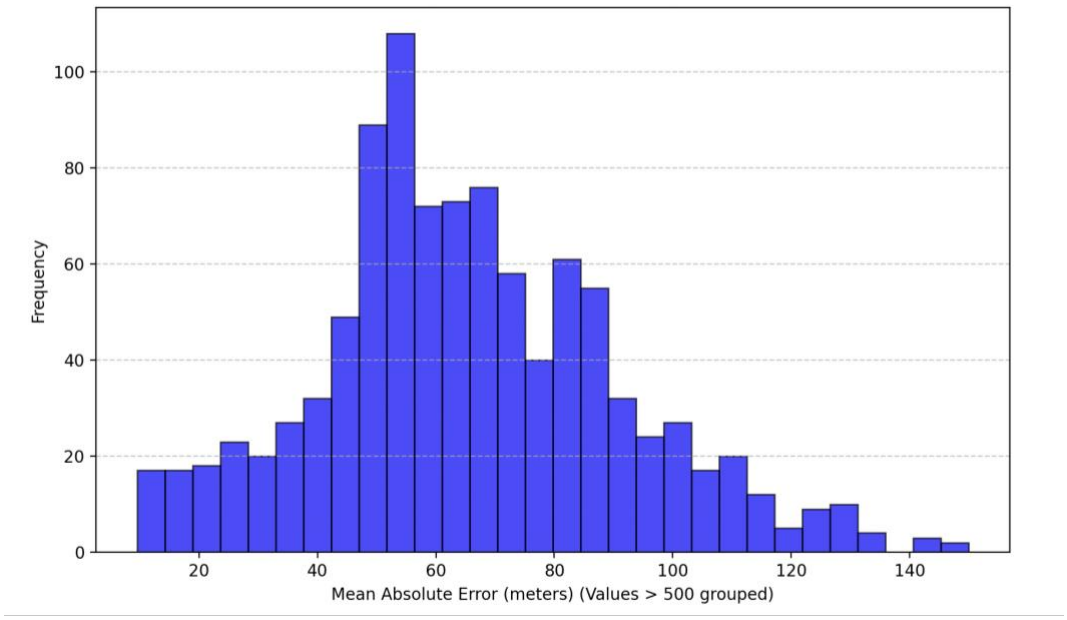


Figure 20. Single mobile attacker with deception - Histogram of Clipped MAE Across Simulations

Figure 20 shows that most simulations result in MAE values ranging between 40 and 100 meters, with a noticeable peak around 50 meters.

This scenario emphasizes the efficacy of deceptive transmission power fluctuations in significantly degrading the attacker's ability to estimate distances. While the majority of simulations result in overestimations, the high variability and extreme outliers demonstrate the robustness of the deception technique.

7.2.5 TWO MOBILE ATTACKER

This scenario introduces two mobile attackers collaborating to localize the transmitter. They share distance estimates and aim to improve their tracking accuracy compared to a single attacker.

Without Deception

The table below summarizes the key metrics for the scenario without deception:

Table 8. Key metrics for two mobile attackers scenario without deception

Metric	Value
Mean Absolute Error (MAE)	3.57m
Root Mean Square Error (RMSE)	5.13m
Standard Deviation Error	3.70m
Minimum error	0.00
25%	0.96m
50%	2.08m
75%	5.09m
Maximum error	24.76m

- The Mean Absolute Error (MAE) of 3.57 meters indicates high localization accuracy, with both attackers managing to stay close to the transmitter during the simulations.
- The Root Mean Square Error (RMSE) of 5.13 meters, slightly higher than the MAE, highlights occasional larger deviations in some simulations, though these remain moderate.

- The Standard Deviation Error of 3.68 meters demonstrates low variability across simulations, suggesting consistent performance in tracking the transmitter.
- The percentiles show that half of the errors are below 2.08 meters, and 75% of the errors are under 5.09 meters, further emphasizing reliable performance.

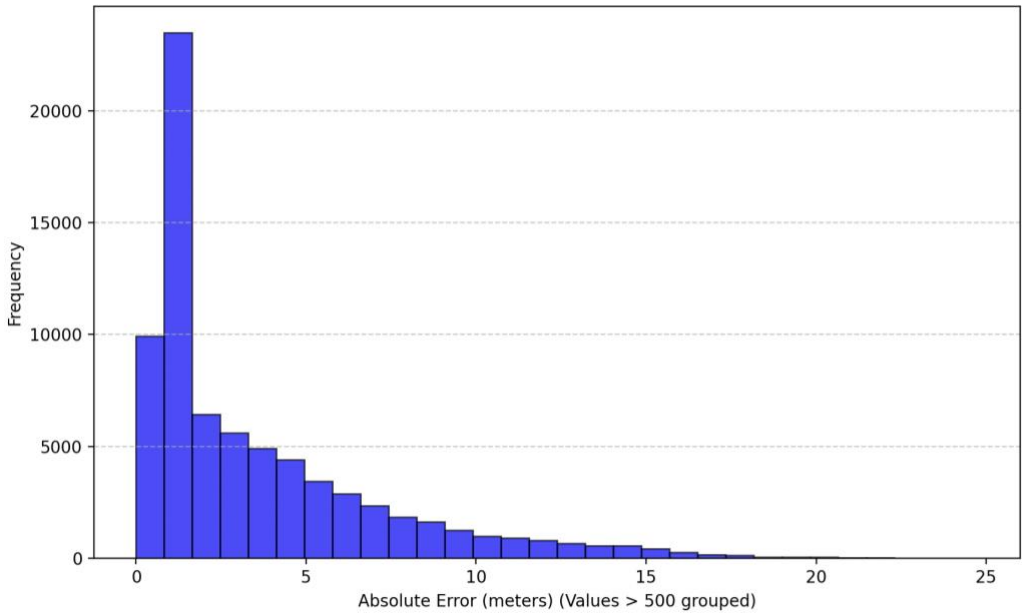


Figure 21. Two mobile attackers without deception - Histogram of clipped absolute errors across all simulations

Figure 21 illustrates the distribution of absolute errors across all simulations. Most errors are concentrated below 10 meters, showcasing consistent and accurate tracking by the attackers.

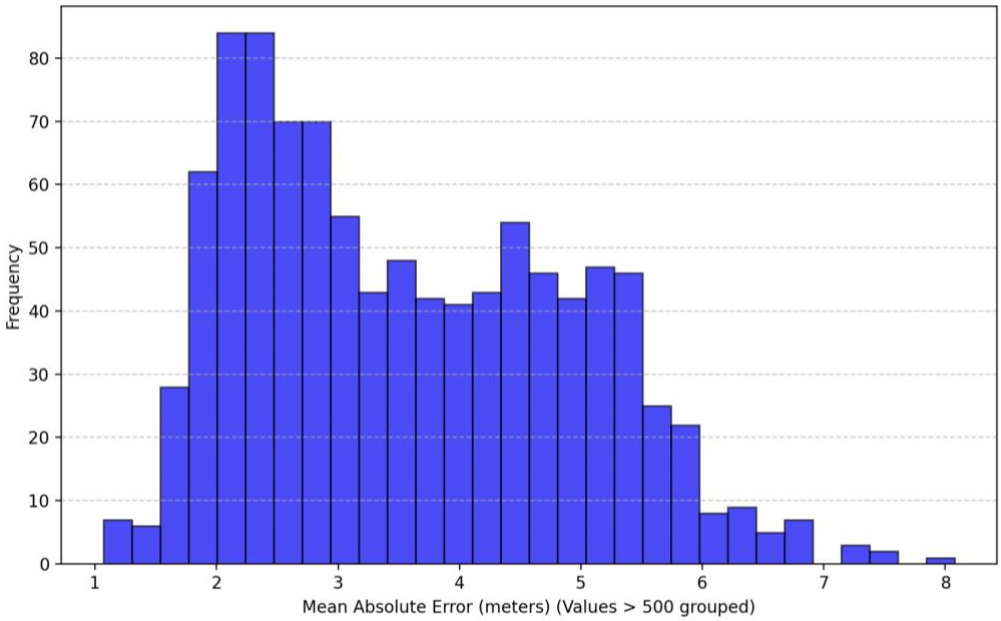


Figure 22. Two mobile attackers without deception - Histogram of Clipped MAE Across Simulations

Figure 22 presents the distribution of the Mean Absolute Error (MAE) across simulations. The majority of simulations exhibit MAE values within the 2-5 meter range, indicating stable performance across trials.

With Deception

Table 9 summarizes the key metrics obtained for this scenario:

Table 9. Key metrics for two mobile attackers scenario with deception

Metric	Value
Mean Absolute Error (MAE)	51.52m
Root Mean Square Error (RMSE)	65.11m
Standard Deviation Error	39.82m

Minimum error	1.12m
25%	22.57m
50%	41.25m
75%	69.74m
Maximum error	776.49m

- The Mean Absolute Error (MAE) of 51.52 meters indicates a moderate degree of difficulty for the attackers to converge toward the transmitter's position.
- The Root Mean Square Error (RMSE) of 65.11 meters suggests that larger deviations occur in some simulations, demonstrating the effectiveness of deception in introducing variability.
- A standard deviation of 39.82 meters reflects notable variability in attacker performance, likely influenced by initial positions and the dynamic movement introduced by deceptive power fluctuations.
- Percentile data shows that 75% of the errors lie below 69.74 meters, while the median error (50th percentile) is 41.25 meters, indicating that most simulations result in reasonably close proximity to the transmitter.
- The maximum error of 776.49 meters represents an extreme case where the attackers were significantly misled by the deception mechanism.

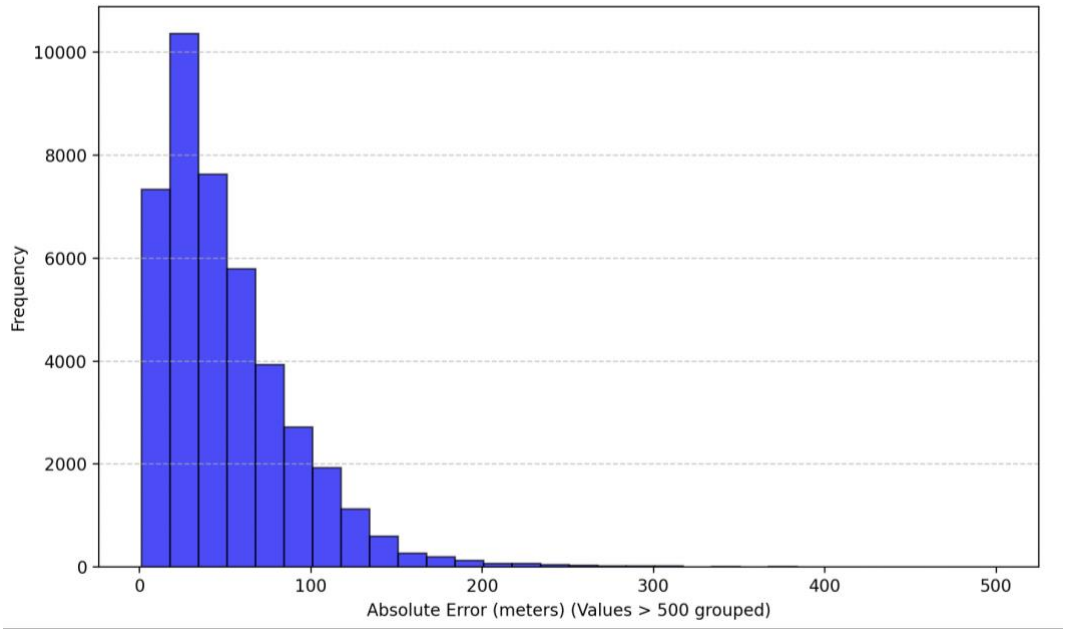


Figure 23. Two mobile attackers with deception - Histogram of clipped absolute errors across all simulations

Figure 23 shows a concentration of errors below 100 meters, with a gradual tapering as errors increase. The small number of outliers (errors exceeding 500 meters) demonstrates the robustness of the deception mechanism in misleading attackers effectively without causing frequent extreme deviations.

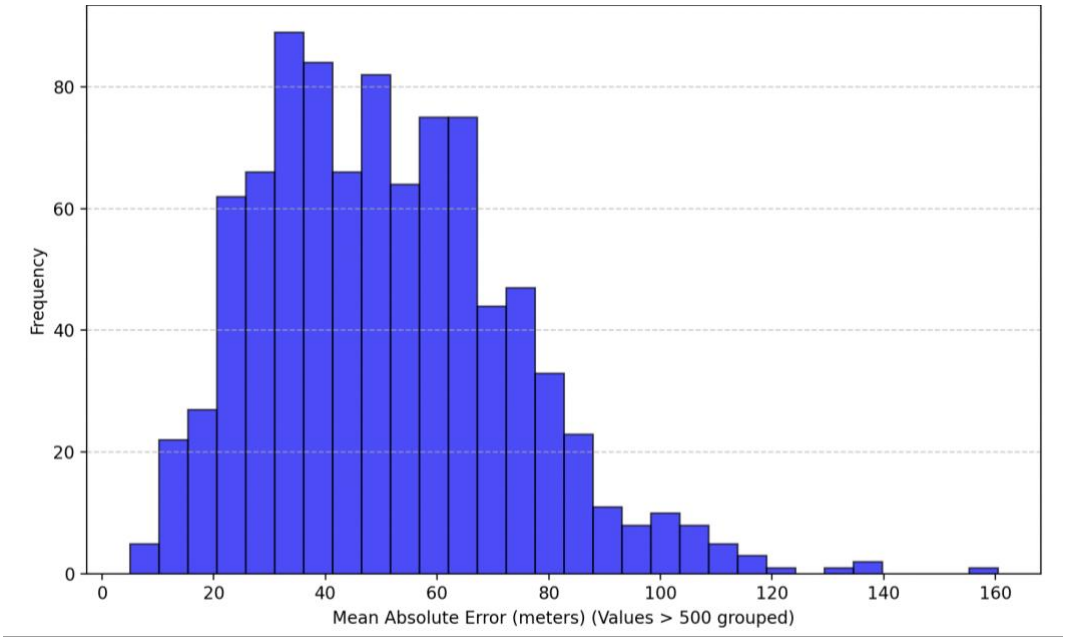


Figure 24. Two mobile attackers with deception - Histogram of Clipped MAE Across Simulations

The distribution of Figure 24 peaks between 30 and 70 meters, with most simulations achieving MAE values within this range. However, the distribution's tail shows that a few simulations resulted in higher MAE values, emphasizing the variability introduced by deceptive power fluctuations.

The results confirm that introducing deception mechanisms significantly impacts the attackers' ability to maintain proximity to the transmitter. The combination of fluctuating power levels and random initial positioning contributes to the observed variability. The low frequency of extreme errors (greater than 500 meters) further indicates the balanced effectiveness of the deception mechanism in this two-mobile-attacker scenario.

7.2.6 THREE MOBILE ATTACKER

This scenario expands the approach to three mobile attackers, enabling the use of trilateration to achieve unique localization of the transmitter.

Without Deception

In a scenario with three mobile attackers and no deception, we would expect the localization error to be near zero, as the attackers can accurately estimate the transmitter's position using precise distance measurements and trilateration.

```
999,20.004,16:13,16:13,1.56387e-11
999,21.001,1:11,1:11,3.6557e-08
999,21.0046,16:13,127.738:-849.461,869.669
999,22.0026,1:11,1:11,2.55389e-08
999,22.0062,16:13,16:13,2.94234e-08
```

Figure 25. Three mobile attackers with deception – Extreme value in log entry

However, the log entry shown in Figure 25, which has the following fields separated by comas ("SimulationId", "Time", "TransmitterPosition", "EstimatedPosition", "Error") indicates an extreme result surrounded by near-zero values, which is likely due to:

1. Numerical instability in trilateration: the attackers' positions during the simulation may have been nearly collinear, leading to instability in the trilateration algorithm. Such configurations can produce unrealistic results when small denominators are encountered in the calculations.
2. Edge cases in initialization or random seed: the random positioning or movement of attackers might have created an edge case where the trilateration logic failed to handle the geometry properly, resulting in an extreme or nonsensical output.

While most simulations align with expectations, with errors close to zero, this behavior highlights occasional anomalies. The fact that only 32 values out of 72,206 exceed 500

meters demonstrates that such extreme outliers are rare but emphasize the need for safeguards against numerical or geometric instabilities in the localization algorithm.

With Deception

The table below summarizes the key metrics obtained for this scenario:

Table 10. Key metrics for three mobile attackers scenario with deception

Metric	Value
Mean Absolute Error (MAE)	915.43m
Root Mean Square Error (RMSE)	19,190.04m
Standard Deviation Error	19,168.33m
Minimum error	0.01m
25%	24.23m
50%	61.18m
75%	142.61m
Maximum error	2,249,280m

- Mean Absolute Error (MAE): 915.43 meters, indicating significant difficulty for the attackers in closing the distance to the transmitter due to deception.
- Root Mean Square Error (RMSE): 19,190.04 meters, showcasing the presence of substantial deviations in many simulations, driven by the deceptive mechanism.
- Standard Deviation: 19,168.33 meters, revealing considerable variability in performance across different runs, largely influenced by random initial positions and coordinated movements.

- Percentiles:
    - 50% (Median): 61.18 meters, meaning half of the simulations resulted in errors below this threshold.
    - 75%: 142.61 meters, illustrating that most errors remain under this range, but extreme outliers exist.
    - Maximum Error: 2,249,280 meters, representing a scenario where the deception mechanism completely misled the attackers.

Additional observations for extreme errors show the following:

- Total values exceeding 500 meters: 5035 out of 73,292, showcasing the frequency of extreme cases where attackers were entirely misdirected.
- Per simulation: 122 out of 73,292, emphasizing the presence of such deviations in a subset of trials.

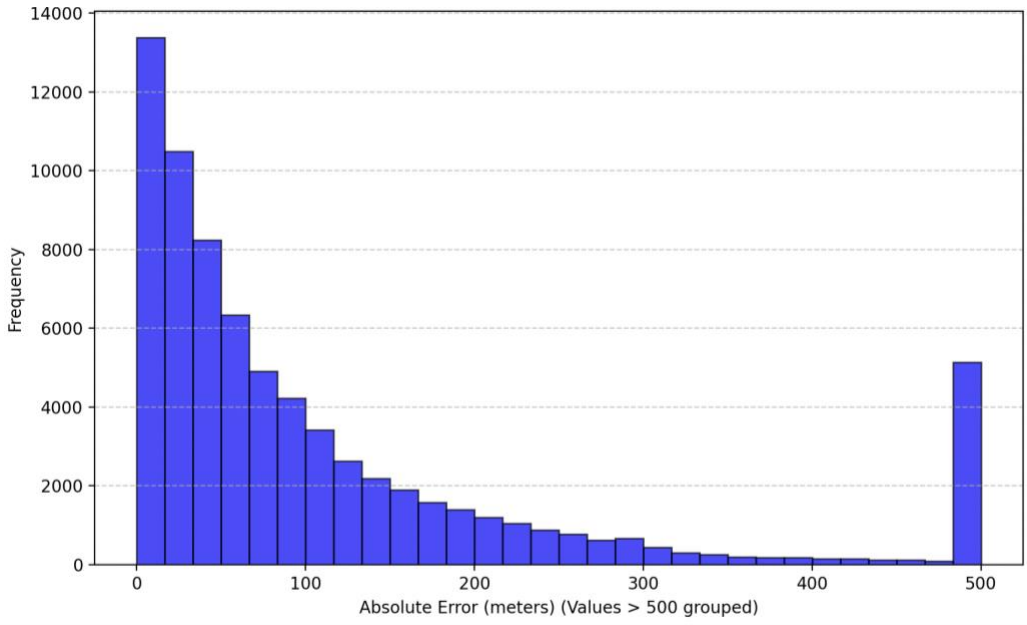


Figure 26. Three mobile attackers with deception - Histogram of clipped absolute errors across all simulations

Figure 26 demonstrates that most errors are concentrated below 200 meters, but the long tail extending toward 500 meters reflects cases in which deception mechanisms severely impacted localization accuracy. The peak at the clipped 500-meter mark underscores instances where attackers were completely diverted.

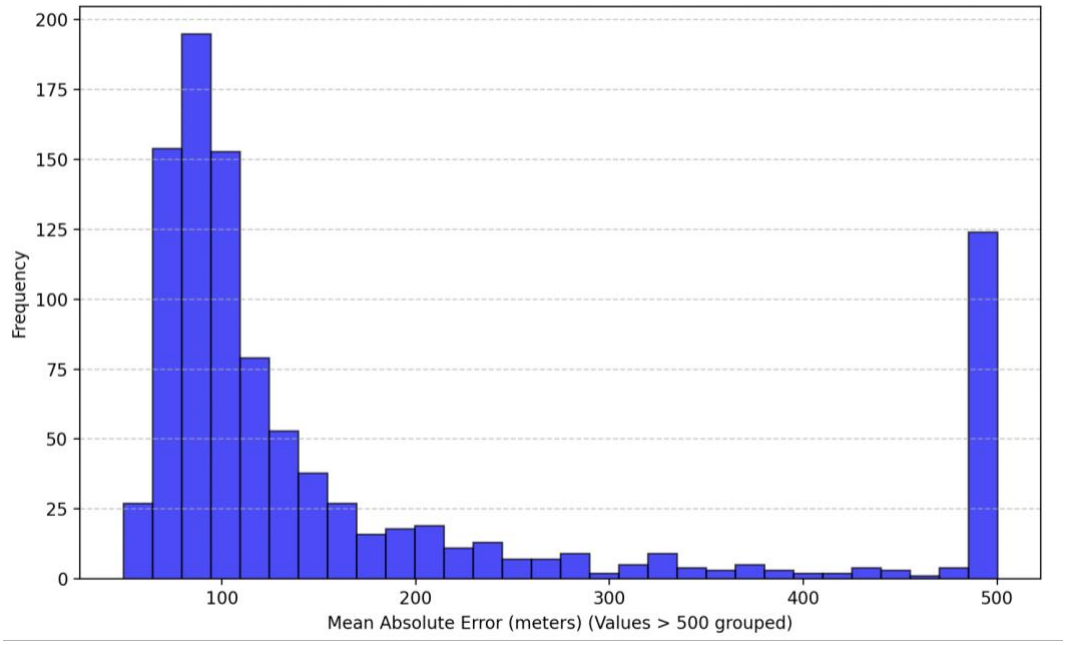


Figure 27. Three mobile attackers with deception - Histogram of Clipped MAE Across Simulations

Figure 27 shows a concentration of simulations achieving MAE values between 50 and 150 meters, with the distribution tailing off at higher values. The spike at the upper limit highlights the variability in attacker performance in the clipped 500-meter value.

This scenario demonstrates the overwhelming effectiveness of the deception mechanism in obstructing the ability of three coordinated mobile attackers to reliably converge toward the transmitter. The substantial MAE and RMSE values, coupled with the presence of extreme outliers, highlight the robustness of deceptive power fluctuations.

7.3 COMPARATIVE ANALYSIS

The comparative analysis evaluates the localization performance across all scenarios, focusing on the effects of deception mechanisms, the number of attackers, and the impact of mobility. The key observations are summarized as follows.

Impact of Deception Mechanisms

The introduction of deception mechanisms consistently degraded localization accuracy across all scenarios. This is evidenced by significantly higher Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) values in scenarios with deception compared to those without:

- For single attackers (static and mobile), the MAE increased from near-zero values without deception to substantial errors when deception was introduced. For example, the single static attacker's MAE rose from effectively 0m to 12.82m with deception.
- In multi-attacker scenarios, the increased coordination capability did not mitigate the impact of deception. For instance, in the three mobile attackers' scenario, the MAE surged to 915.43m, highlighting the effectiveness of deceptive power fluctuations in misleading even coordinated attackers.

Effect of Increasing the Number of Attackers

Adding more attackers improved localization accuracy in scenarios without deception due to the redundancy and increased precision from multiple perspectives:

- For three static attackers without deception, the MAE reduced to 6m compared to 0m for a three static attackers due to computing a unique intersection point, avoiding the ambiguity of two solutoions in the two attacker scenario. In contrast, for a single static attacker without deception, the localization results in a continuous circle of possible positions rather than a discrete set of points, making direct comparisons with multi-attacker scenarios less meaningful.

- However, with deception, the benefits of multiple attackers diminished as the coordinated localization strategies became susceptible to error amplification. This is evident in the three mobile attackers' scenario, where the deception mechanism caused extreme localization errors, with the RMSE exceeding 19,000m.

Role of Mobility

Mobility introduced dynamic behavior that allowed attackers to refine their positions over time, offering advantages in certain scenarios.

- Without deception, the single mobile attacker achieved an MAE of 0.01m, which is comparable to the perfect localization accuracy (MAE = 0) achieved by the single static attacker. However, the introduction of mobility allowed for continuous adjustments, demonstrating the potential for adaptability.
- With deception, this advantage diminished as variability in RSSI-based distance estimations introduced significant errors. For instance, the single mobile attacker's MAE increased to 65.47m with deception, highlighting the difficulty of maintaining accurate tracking in the presence of fluctuating RSSI values.

Outlier Analysis

The analysis of extreme errors (>500m) revealed interesting patterns:

- Scenarios with three attackers (static and mobile) exhibited the highest number of extreme errors under deception, with 5035 out of 73,292 values exceeding 500m in the three mobile attackers' case.
- In contrast, scenarios with fewer attackers or without deception rarely encountered such extreme errors, highlighting the compounded impact of deception and coordination complexity in multi-attacker setups.

Final Comparative Box Plot Across All Scenario

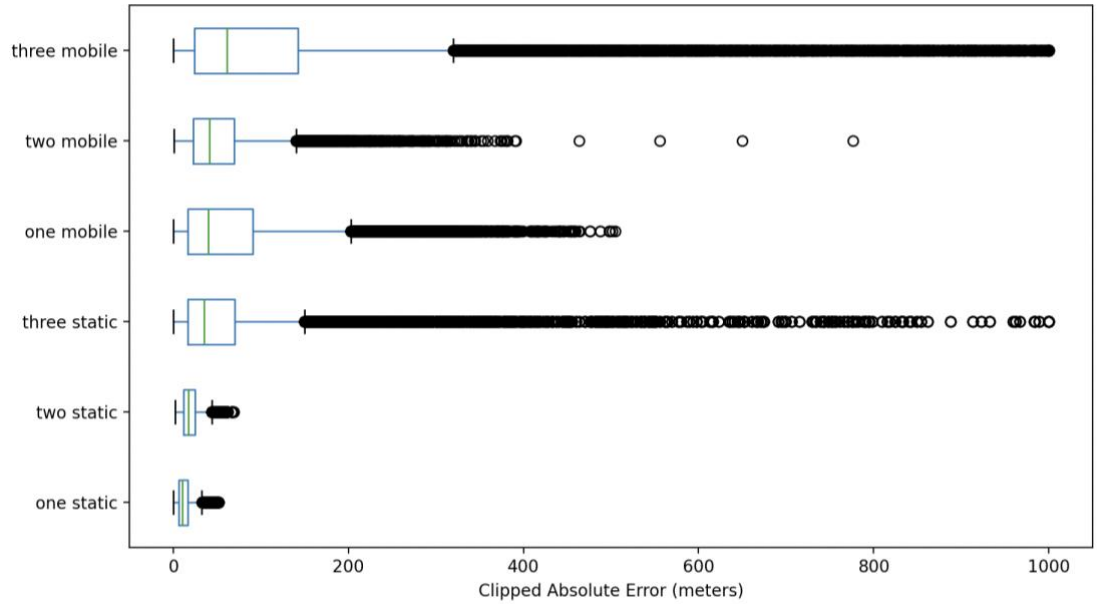


Figure 28. Box plot of clipped absolute error by scenario

The box plot in Figure 28 highlights the impact of deception mechanisms on localization accuracy across different scenarios, showing clear trends as the complexity of the setup increases. For simpler cases, such as a single static attacker and two static attackers, errors remain tightly distributed with narrow interquartile ranges (IQRs) and minimal outliers. These scenarios demonstrate consistent localization performance, as the limited number of attackers minimizes the propagation of errors caused by deceptive fluctuations in transmission power.

In more complex scenarios, such as three static attackers and mobile attackers, the error distributions widen significantly. This is especially evident in the three mobile attacker scenario, where coordinated trilateration amplifies the effects of deceptive RSSI values. Small power fluctuations in these scenarios cause larger inaccuracies, as multiple attackers rely on intersecting distance circles that are more prone to distortion when errors occur.

Mobility further compounds these issues, as attackers dynamically adjust their positions based on misleading RSSI trends, leading to greater variability in localization accuracy.

The most pronounced impact of deception is seen in the three mobile attacker scenario, where the highest median errors and the greatest number of extreme outliers occur. This underscores the susceptibility of coordinated and mobile attackers to compounded errors introduced by deceptive mechanisms. Overall, the box plot demonstrates that as the number of attackers and their reliance on shared or dynamic strategies increase, the effectiveness of deception grows, significantly disrupting their ability to localize the transmitter accurately.

7.4 SUMMARY

This section summarizes the key findings from the analysis of results across all scenarios, highlighting the effectiveness of deception mechanisms and their impact on localization accuracy for static and mobile attackers.

1. Static Attackers

- In scenarios without deception, static attackers achieve high localization accuracy due to the absence of interference in RSSI measurements. The Mean Absolute Error (MAE) is effectively 0m for a single static attacker since the attacker can determine the exact distance but not a unique position. For two static attackers, the MAE is 6m, as their localization produces two possible solutions, with one being correct and the other introducing some error. For three static attackers, the MAE is close to 0m, as trilateration enables a unique and precise estimation of the transmitter's position.
- Deception mechanisms significantly degrade accuracy, increasing MAE and introducing greater variability. For a single static attacker, MAE increased to 12.82m, while for two and three static attackers, MAE rose to 19.41m and 80.83m, respectively. The results reveal that more attackers are increasingly susceptible to power fluctuations due to compounded errors in their calculations, particularly in trilateration.

2. Mobile Attackers

- Without deception, mobile attackers demonstrate near-perfect or highly accurate localization due to their ability to refine their positions over time based on RSSI measurements. For example, a single mobile attacker achieves an MAE of 0.01m, while two mobile attackers maintain accuracy with an MAE of 3.57m.
- Deception introduces substantial inaccuracies, particularly in coordinated scenarios. A single mobile attacker's MAE increased to 65.47m, while two and three mobile attackers experienced MAEs of 51.52m and 915.43m, respectively. The latter demonstrates the compounded effect of deceptive power fluctuations on trilateration and coordinated movements, resulting in extreme outliers and high variability.

3. Comparative Analysis

- The comparative box plot reveals that simpler scenarios with fewer attackers and static setups (e.g., one or two static attackers) are less impacted by deception. These setups exhibit narrower error distributions and fewer extreme outliers.
- In contrast, more complex scenarios with coordinated or mobile attackers (e.g., three mobile attackers) are highly susceptible to deception, showing wider interquartile ranges and a greater prevalence of extreme errors. Trilateration-based methods and dynamic movements amplify the impact of deceptive RSSI fluctuations, causing significant deviations in localization accuracy.

8. CONCLUSIONS & FUTURE WORK

Conclusions

This study has demonstrated the effectiveness of deception mechanisms, specifically power fluctuations, in degrading the localization performance of both static and mobile attackers in wireless sensor networks. The findings reveal that while static attackers achieve high localization accuracy in the absence of deception, their performance degrades as power fluctuations introduce uncertainty into RSSI-based distance estimations. Mobile and coordinated attackers, despite their ability to refine their positions dynamically, become increasingly vulnerable to deception as errors in their distance estimations compound over time. This highlights a key conclusion: while more sophisticated attackers have the potential for greater accuracy, they are also more susceptible to deception-induced variability, particularly in coordinated scenarios where errors are amplified due to compounding miscalculations in their shared estimations.

From a practical perspective, implementing controlled power fluctuations poses challenges, such as ensuring synchronization across the network and maintaining energy efficiency. The simulation framework, while robust, operates under certain constraints, such as assumptions about attacker behavior, idealized environmental conditions, and limitations in computational resources. These constraints may limit the direct applicability of the results to real-world scenarios, where attackers might employ more adaptive or memory-based strategies, and environmental factors like multipath propagation could affect RSSI measurements.

Future Work

To further enhance the efficacy of deception mechanisms and address the limitations observed in this study, several avenues for future work are proposed:

1. **Luring zones with synchronized power fluctuations**: introducing luring zones, where nodes collaborate to create controlled and synchronized power fluctuations, could

actively mislead attackers into specific areas. This strategy could further enhance the network's ability to disrupt coordinated attackers by exploiting their reliance on trilateration or movement patterns.

2. **Incorporating more complex attacker behaviors**: future research could explore attackers with advanced strategies, such as clustering node locations or incorporating memory to store and analyze previous locations. This would present a more realistic challenge, requiring the deception mechanisms to adapt dynamically to more intelligent adversaries.
3. **Optimization of fluctuation patterns**: research could focus on optimizing the fluctuation patterns, including the amplitude, frequency, and duration of power changes, to maximize their disruptive impact while minimizing the energy cost to the network. The timing of these fluctuations could also be synchronized with anticipated attacker behavior to amplify their effectiveness.
4. **Real-world implementation and validation**: extending the work to real-world deployments would validate the practical feasibility of these mechanisms. This would involve addressing hardware constraints, environmental noise, and the challenges of synchronization across a distributed network in a physical setting.

This work provides a foundational step in the study of deception mechanisms for network security, highlighting their potential to protect wireless sensor networks against increasingly sophisticated attackers.

9. REFERENCES

[1] State of IoT 2024: Number of connected IoT devices growing 13% to 18.8 billion globally. Retrieved the 10th of January 2025 from https://iot-analytics.com/number-connected-iot-devices/

[2] Zscaler ThreatLabz Finds a 400% Increase in IoT and OT Malware Attacks Year-over-Year. Retrieved the 10th of January 2025 from https://www.zscaler.com/press/zscaler-threatlabz-finds-400-increase-iot-and-ot-malware-attacks-year-over-year-underscoring

[3] The 2024 Benchmark Report on IoT Security. Retrieved the 10th of January 2025 from https://www.paloaltonetworks.com/apps/pan/public/downloadResource?pagePath=/content/pan/en_US/resources/research/the-2024-benchmark-report-on-iot-security

[4] ns (simulator). Retrieved the 11th of January 2025 from https://en.wikipedia.org/wiki/Ns_(simulator)

[5] What is ns-3. Retrieved the 11th of January 2025 from https://www.nsnam.org/about/what-is-ns-3/

[6] NetAnim. Retrieved the 11th of January 2025 from https://www.nsnam.org/wiki/NetAnim

[7] "IEEE Standard for Low-Rate Wireless Networks," in IEEE Std 802.15.4-2020 (Revision of IEEE Std 802.15.4-2015) , vol., no., pp.1-800, 23 July 2020, doi: 10.1109/IEEESTD.2020.9144691.

[8] IEEE 802.15.4. Retrieved the 12th of January 2025 from https://en.wikipedia.org/wiki/IEEE_802.15.4

[9] Introduction of IEEE 802.15.4 Technology. Retrieved the 12th of January 2025 from https://www.geeksforgeeks.org/introduction-of-ieee-802-15-4-technology/

[10] G. Montenegro, N. Kushalnagar, J. Hui, and D. Culler, “Transmission of IPv6 packets over IEEE 802.15.4 networks,” IETF RFC 4944 (Standards Track), Sept. 2007

[11] What is 6LoWPAN?. Retrieved the 12th of January 2025 from https://www.geeksforgeeks.org/what-is-6lowpan/

[12] 6LoWPAN. Retrieved the 12th of January 2025 from https://en.wikipedia.org/wiki/6LoWPAN

[13] Y. Ebrahimi and M. Younis, "Traffic Analysis Through Spatial and Temporal Correlation: Threat and Countermeasure," in IEEE Access, vol. 9, pp. 54126-54151, 2021, doi: 10.1109/ACCESS.2021.3070841.

[14] Y. Tan, J. Liu and J. Wang, "How to Protect Key Drones in Unmanned Aerial Vehicle Networks? An SDN-Based Topology Deception Scheme," in IEEE Transactions on Vehicular Technology, vol. 71, no. 12, pp. 13320-13331, Dec. 2022, doi: 10.1109/TVT.2022.3200339.

[15] Q. He, S. Fang, T. Wang, Y. Liu, S. Zhao and Z. Lu, "Proactive Anti-Eavesdropping With Trap Deployment in Wireless Networks," in IEEE Transactions on Dependable and Secure Computing, vol. 20, no. 1, pp. 637-649, 1 Jan.-Feb. 2023, doi: 10.1109/TDSC.2022.3141406.

[16] B. Alkanjr and I. Mahgoub, "A Novel Deception-Based Scheme to Secure the Location Information for IoBT Entities," in IEEE Access, vol. 11, pp. 15540-15554, 2023, doi: 10.1109/ACCESS.2023.3244138.

[17] I. Butun and I. Mahgoub, "Expandable Mix-Zones as a Deception Technique for Providing Location Privacy on Internet-of-Battlefield Things (IoBT)," 2024 International Conference on Smart Applications, Communications and Networking (SmartNets), Harrisonburg, VA, USA, 2024, pp. 1-7, doi: 10.1109/SmartNets61466.2024.10577689.

[18] J. Choi, J. Ha and H. Jeon, "Physical layer security for wireless sensor networks," *2013 IEEE 24th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, London, UK, 2013, pp. 1-6, doi: 10.1109/PIMRC.2013.6666094.

[19] I. Marjanović, D. Milić, J. Anastasov, and A. Cvetković, "Physical layer security of wireless sensor network based on opportunistic scheduling," *Facta Universitatis, Series: Automatic Control and Robotics*, vol. 19, no. 1, pp. 1-10, 2020. doi: 10.22190/FUACR2001001M

[20] ns-3 Tutorial. Retrieved the 12th of January 2025 from https://www.nsnam.org/docs/release/3.37/tutorial/singlehtml/index.html

[21] ns-3.37. Retrieved the 12th of January 2025 from https://www.nsnam.org/releases/ns-3-37/

[22] Sustainable Development Goals. Retrieved the 10th of January 2025 from https://www.un.org/sustainabledevelopment/

[23] Azote Images for Stockholm Resilience Centre (CC BY 4.0)

[24] The SDGs wedding cake. Retrieved the 10th of January 2025 from

https://www.stockholmresilience.org/research/research-news/2016-06-14-the-sdgs-wedding-cake.html

[25] Sustainable Development Goals: "better model for the future". Retrieved the 5th of July from https://blog.dgnb.de/en/sdgs-interview-buckley-part-1/

[26] United Nations - THE 17 GOALS. Retrieved the 10th of January 2025 from https://sdgs.un.org/goals

[27] SDG Tracker: Measuring progress towards the Sustainable Development Goals. Retrieved the 10th of January 2025 from **https://ourworldindata.org/sdgs**

ANNEX I: INTEGRATION OF THE SDGS INTO THE PROJECT

The Sustainable Development Goals (SDGs) are a set of 17 global objectives designed to achieve a more sustainable future. They were adopted by the United Nations in 2015 as a universal call for action by all countries to promote prosperity while protecting the planet [22]. The SDGs aim to address the global challenges we face, including poverty, climate change, peace, and justice, among others. They are intended to be achieved by 2030.

Shortly before the official publication of the SDGs, Professor Rockstrom, a recognized scientist for his work on global sustainability, and partners presented an interesting model, the "wedding cake" model for the SDGs. The diagram that represents this model is shown in Figure 29.

Figure 29 - The wedding cake model for SDGs diagram [23]

The pie model describes how economies and societies should be seen as embedded parts of the biosphere. This approach differs from the current vision in which the social, economic, and ecological areas are seen as independent. An integrated view of social, economic, and ecological development is supported by such a conception [24].

In other words, the illustration shows that the basis of all Sustainable Development Goals is the biosphere, which provides all the resources we need to live. We should do everything in our power to maintain it, to meet the basic needs of the next level, society. Similarly, society is the basis for a well-functioning economy [25].

Although it may seem difficult to relate a software-related project with sustainable goals, if we think more deeply, the ultimate goal of the project is to improve the security of low-power wireless networks, essential pillars of contemporary infrastructures. The security techniques presented in the project at the physical layer show that this research directly aligns with some of the SDGs due to the wide range of scenarios in which wireless sensor networks can be used, including environmental monitoring (e.g., tracking air and water quality, deforestation, and wildlife movement), disaster management (e.g., early-warning systems for floods, earthquakes, and wildfires), healthcare (e.g., remote patient monitoring and emergency response systems), smart cities (e.g., traffic management, energy-efficient buildings, and waste management), industrial automation (e.g., predictive maintenance and resource monitoring), and agriculture (e.g., precision farming and irrigation control).

With that being explained, the SDGs that are more related to this project are shown in Table 11:

SDG identidied	SDG Dimenssion	Role
9 - Industry, innovation and infrastructure	Economy	Primary
11 – Sustainable cities and communities	Society	Primary
13 – Climate action	Biosphere	Primary

3 – Good health and well-being	Society	Secondary

Table 11. SDGs integrated into this project. SDGs obtained from [28]

Additionally, a more detailed explanation of how the project aligns with the SDGs and the specific targets it addresses are presented below [26][27].

SDG #9 – Industry, innovation and infrastructure

SDG 9 focuses on building resilient infrastructure, promoting inclusive and sustainable industrialization, and fostering innovation. By securing wireless communications, this project improves the reliability and security of these networks and makes them more resilient to attackers.

In particular, the goal this project achieves is target 9.1: Develop sustainable, resilient and inclusive infrastructures. The UN has defined it as: "Develop quality, reliable, sustainable and resilient infrastructure, including regional and transborder infrastructure, to support economic development and human well-being, with a focus on affordable and equitable access for all."

Strengthening wireless communications at the physical layer ensures that critical industrial and supply chain networks can withstand malicious attempts of disruptions, contributing both to access to essential services. The reliability of data flows within sensors is crucial to maintain stable productivity levels, even in resource-constrained regions, with the final objective of fostering innovation in industrial processes.

SDG #11 – Sustainable cities and communities

The functionality of smart cities is heavily dependent on WSNs for effective resource management, transportation systems, and public safety. By defending WSNs against physical-layer attacks, our research creates a safer and more resilient urban environment by ensuring the continuity and security of vital services. Specifically, this project tackles several targets within this SDG, which are target 11.2, 11.4 and 11.5.

Target 11.2 aims to "Provide access to safe, affordable, accessible and sustainable transport systems for all", which relies heavily on reliable real-time data from traffic and road sensors. Additionally, target 11.4 promotes "protecting and safeguarding the world's cultural and natural heritage", well-secured environmental sensor networks can help monitor historic sites and protected ecological areas against environmental threats or even unauthorized intrusions. Finally, target 11.5 focuses on "significantly reducing the number of deaths and the number of people affected by disasters, which is supported by robust wireless sensor infrastructures that continue to function during floods, fires or earthquakes, even under adversarial conditions. These systems provide critical data for timely evacuations and interventions and can also aid in disaster prediction, enabling better preparation

SDG #13 – Climate action

Ad hoc wireless networks are usually utilized in environmental monitoring systems to tackle climate-related challenges, such as pollution tracking and natural resource management. By securing these networks, this project ensures the integrity and reliability of the data they collect, guiding climate-related decision-making.

Related to the last-mentioned target of SDG #11, this project integrates target 13.1 by "strengthening resilience and adaptive capacity to climate-related hazards and natural disasters in all countries". Reliable sensor data is a crucial asset for anticipating extreme weather and mitigating their impact. Preventing adversaries from tampering with or intercepting climate-monitoring signals preserves the accuracy of early-warning systems and facilitates better coordinated responses to environmental crises.

SDG #3 – Good health and well-being

Numerous healthcare applications depend on Internet of Things (IoT) and Wireless Sensor Network (WSN) technologies, including remote patient monitoring, emergency response systems, and health surveillance initiatives. By enhancing the security of communication channels utilized in these essential applications, protecting sensitive data, and ensuring the dependability of healthcare systems, this project aligns with SDG 3.

In particular, target 3.6 aims to "halve the number of global deaths and injuries from road traffic accidents", which is heavily dependent on accurate sensor networks for traffic flow control and rapid emergency responses. Also, target 3.9 addresses "substantially reducing the number of deaths and illnesses from hazardous chemicals and air, water and soil pollution

and contamination.", an objective that is directly related to trustworthy monitoring efforts that ensure that health officials can act on reliable information and, thus, protecting communities from environmental health risks.

ANNEX II: DEV/BUILD TOOL INFORMATION

1. **System Information**
    a. Operating System and Version: macOS Sequoia 15.1.1
    b. Hardware Platform: Apple Silicon M1 Pro (16 GB RAM)
2. **Compiler Toolchain**
    a. Compiler Name and Version: Clang 16.0.0
    b. Language Standard: C++17
    c. Build Tools:
        i. Cmake version: 3.30.5
        ii. Ninja version: 1.12.1
3. **Software Dependencies and Versions**
    a. ns-3 version: ns-3.37
    b. NetAnim version: netanim-3.108 (for visualizing topology and mobility)
    c. Python and plotting:
        i. Python 3.12.4
        ii. matplotlib 3.8.4

ANNEX III: MONTE CARLO METHOD SCRIPT FOR RANDOM NETWORK TOPOLOGY GENERATION

```
#!/bin/bash

NUM_SIMULATIONS=1000
SCENARIO="09XmobileTripleCoordinatedDeception"
NUM_ATTACKERS=3

./ns3 clean
./ns3 configure

for i in $(seq 1 $NUM_SIMULATIONS); do
    i=$(printf "%d" $i) # Ensure i is treated as an integer

    SINK_X=$(awk -v min=0 -v max=20 -v seed=$RANDOM 'BEGIN{srand(seed); print
min+rand()*(max-min)}')
    SINK_Y=$(awk -v min=0 -v max=20 -v seed=$RANDOM 'BEGIN{srand(seed); print
min+rand()*(max-min)}')
    SENSOR_X=$(awk -v min=0 -v max=20 -v seed=$RANDOM 'BEGIN{srand(seed); print
min+rand()*(max-min)}')
    SENSOR_Y=$(awk -v min=0 -v max=20 -v seed=$RANDOM 'BEGIN{srand(seed); print
min+rand()*(max-min)}')
    ATTACKER_POSITIONS=()
    for (( j=0; j<$NUM_ATTACKERS; j++ )); do
        AX=$(awk -v min=0 -v max=20 -v seed=$RANDOM 'BEGIN{srand(seed); print
min+rand()*(max-min)}')
        AY=$(awk -v min=0 -v max=20 -v seed=$RANDOM 'BEGIN{srand(seed); print
min+rand()*(max-min)}')
        ATTACKER_POSITIONS+=("$AX" "$AY")
    done

    ATTACKER_ARGS=""
    for (( j=0; j<$NUM_ATTACKERS; j++ )); do
        AX=${ATTACKER_POSITIONS[$((j * 2))]}
        AY=${ATTACKER_POSITIONS[$((j * 2 + 1))]}
        ATTACKER_ARGS+=" --attackerX$j=$AX --attackerY$j=$AY"
    done

    ./ns3 run "scratch/$SCENARIO --simulationId=$i --sinkX=$SINK_X --
sinkY=$SINK_Y$ATTACKER_ARGS --sensorX=$SENSOR_X --sensorY=$SENSOR_Y" > /dev/null
2>&1

    echo "Scenario $SCENARIO - Simulation $i complete: Sink=($SINK_X, $SINK_Y),
Sensor=($SENSOR_X, $SENSOR_Y), Attackers=$ATTACKER_ARGS"
done
```