

UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS
CARRERA DE TELECOMUNICACIONES



TEMA:

SISTEMA DE DETECCIÓN DE INTRUSOS APLICANDO INTELIGENCIA ARTIFICIAL PARA LA DETECCIÓN DE ATAQUES EN UNA RED DE SENSORES INALÁMBRICOS (WSN).

**TRABAJO DE GRADO PREVIO A LA OBTENCIÓN DEL TÍTULO DE
INGENIERÍA DE TELECOMUNICACIONES**

AUTOR:

JHOSELYN LIZETH VELASTEGUI MORALES

DIRECTOR:

ING. FABIÁN GEOVANNY CUZME RODRÍGUEZ, MSC.

Ibarra, 2024



UNIVERSIDAD TÉCNICA DEL NORTE

BIBLIOTECA UNIVERSITARIA

AUTORIZACIÓN DE USO Y PUBLICACIÓN

A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

1. IDENTIFICACIÓN DE LA OBRA

En cumplimiento del Art. 144 de la Ley de Educación Superior, hago la entrega del presente trabajo a la Universidad Técnica del Norte para que sea publicado en el Repositorio Digital Institucional, para lo cual pongo a disposición la siguiente información:

DATOS DEL CONTACTO			
CÉDULA DE IDENTIDAD	1003607767		
APELLIDOS Y NOMBRES	Velasgui Morales Jhoselyn Lizeth		
DIRECCIÓN	Otavalo, Antonio Estévez Mora y Av. Atahualpa		
E-MAIL	jlvelasteguim@utn.edu.ec		
TELÉFONO FIJO	062920396	TELÉFONO MÓVIL	0967260907

DATOS DE LA OBRA			
TÍTULO	SISTEMA DE DETECCIÓN DE INTRUSOS APLICANDO INTELIGENCIA ARTIFICIAL PARA LA DETECCIÓN DE ATAQUES EN UNA RED DE SENSORES INALÁMBRICOS (WSN)		
AUTOR	Velasgui Morales Jhoselyn Lizeth		
FECHA	18-01-2024		
PROGRAMA	X	PREGRADO	POSTGRADO
TÍTULO	Ingeniera en Telecomunicaciones		
DIRECTOR	Ing. Cuzme Rodríguez Fabián Geovanny, Msc		



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

2. CONSTANCIAS

El autor manifiesta que la obra objeto de la presente autorización es original y se la desarrolló, sin violar derechos de autor de terceros, por lo tanto, la obra es original y que es el titular de los derechos patrimoniales, por lo que asume la responsabilidad sobre el contenido de la misma y saldrá en defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra, a los 18 días del mes de enero de 2024

EL AUTOR

Velastegui Morales Jhoselyn Lizeth

CI: 1003607767



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CERTIFICACIÓN

MAGISTER FABIÁN CUZME, DIRECTOR DEL PRESENTE TRABAJO DE TITULACIÓN CERTIFICA:

Que, el presente trabajo de Titulación “SISTEMA DE DETECCIÓN DE INTRUSOS APLICANDO INTELIGENCIA ARTIFICIAL PARA LA DETECCIÓN DE ATAQUES EN UNA RED DE SENSORES INALÁMBRICOS (WSN)” ha sido desarrollado por la señorita Velastegui Morales Jhoselyn Lizeth bajo mi supervisión.

Es todo en cuanto puedo certificar en honor de la verdad.


 FABIAN GEOVANNY
CUZME RODRIGUEZ

Ing. Fabián Geovanny Cuzme Rodríguez, MsC.

C.I. 1311527012

DIRECTOR

DEDICATORIA

Dedico este trabajo de titulación a mi familia, que estuvieron a mi lado de manera constante, ofreciéndome su respaldo incondicional en todo momento y por creer en mí incluso cuando yo lo dudaba.

A mi madre Mayra que me ha brindado su apoyo a lo largo de todo este proceso, su infinita paciencia, su amor incondicional y sacrificio inmensurable que me han permitido llegar hasta este punto. Gracias, madre por inspirarme y mostrarme la importancia de la perseverancia.

A mis hermanas y sobrinos, quienes, con sus palabras de aliento y motivación constante, han sido soporte esencial a lo largo de este proceso.

Dedico este logro a mi familia que fue mi sistema de apoyo y que me sirvió de guía y me condujo al éxito.

Velastegui Morales Jhoselyn Lizeth

AGRADECIMIENTO

Agradezco sinceramente a mi familia, cuya confianza inquebrantable en mí ha sido el motor de todos mis logros. Sus sacrificios incesantes, sus sabios consejos y su apoyo constante han sido los pilares sobre los que he construido mi camino.

A mis amigos, mis fieles compañeros de viaje, les agradezco su apoyo inquebrantable. Las risas compartidas y los momentos preciosos han hecho que este proceso sea memorable y gratificante.

Quisiera expresar mi sincero agradecimiento a mi tutor, Ing. Cuzme Fabián, por su orientación y apoyo en la elaboración de esta tesis y su dedicación con la formación académica de sus estudiantes.

Estoy profundamente agradecida con todos los que me han acompañado en este viaje y han hecho posible este logro.

INDICE DE CONTENIDO

CAPÍTULO I ANTECEDENTES	1
1.1. Problema.....	1
1.2. Objetivos.....	3
1.2.1. Objetivo General.....	3
1.2.2. Objetivos Específicos	3
1.3. Alcance	3
1.4. Justificación.....	6
CAPÍTULO II MARCO TEÓRICO.....	8
2.1. Red de Sensores Inalámbricos.....	8
2.1.1. Características.....	9
2.1.2. Arquitectura	11
2.1.3. Topologías	15
2.1.4. Aplicaciones	21
2.2. Seguridad de la Información.....	22
2.2.1. Modelo CIA.....	25

2.2.2.	Confidencialidad.....	25
2.2.3.	Integridad.....	27
2.2.4.	Disponibilidad	27
2.3.	Ataques, amenazas y vulnerabilidades a las WSNs	29
2.3.1.	Ataques	29
2.3.2.	Amenazas.....	34
2.3.3.	Vulnerabilidades.....	37
2.4.	Mitigaciones de Ataques a WSNs	38
2.4.1.	SET: Detección de Clones de Nodos.....	38
2.4.2.	Algoritmo para la Detección de Ataque de Sumidero	39
2.4.3.	Detección de Ataques DoS	39
2.5.	Sistema de Detección de Intrusos (IDS).....	40
2.5.1.	Tipos	41
2.5.2.	Componentes Funcionales Fundamentales.....	43
2.5.3.	Softwares	44
2.6.	Inteligencia Artificial.....	49
2.6.1.	Capacidades de la Inteligencia Artificial	50

2.6.2.	Clasificación de la Inteligencia Artificial.....	51
2.6.3.	Algoritmos de Detección.....	53
2.7.	Trabajos Relacionados con Aplicaciones de Inteligencia Artificial.....	55
CAPÍTULO III: DISEÑO DEL IDS		58
3.1.	Requerimientos.....	58
3.1.1.	Especificación de los Requisitos de las Partes Interesadas (StRS)	59
3.1.1.	Especificación de los requisitos del sistema (SyRS)	60
3.1.2.	Especificación de Requisitos Del Software (SRS)	62
3.1.3.	Arquitectura de Requisitos del Sistema (SyRA)	62
3.2.	Selección de Software de Simulación	64
3.3.	Selección de Hardware para Simulación.....	66
3.4.	Selección de Ataque	67
3.4.1.	Identificación de Ataque.....	67
3.5.	Selección de Algoritmo de Detección	69
3.6.	Selección de Sistema Operativo	71
3.7.	Conectividad Inalámbrica Dentro del Entorno de Simulación.....	74
3.7.1.	Distribución de Nodos.....	76

3.7.2. Diseño de la red	83
3.8. Integración del IDS.....	90
3.8.1. Interconexión entre los diferentes componentes al IDS	92
3.9. Integración de inteligencia artificial	94
3.10. Funcionamiento de la configuración del sistema	98
CAPÍTULO IV IMPLMETACIÓN Y PRUEBAS DE FUNCIONAMIENTO.....	101
4.1. Implementación	101
4.2. Pruebas de funcionamiento.....	110
4.2.1. Plan de pruebas.....	111
4.2.2. Pruebas básicas.....	111
4.2.3. Pruebas específicas.....	116
4.3. Discusión	122
CONCLUSIONES.....	125
RECOMENDACIONES	127
BIBLIOGRAFÍA	128
ANEXOS	140
Anexo A.....	140

Anexo B..... 145

Anexo C..... 151

ÍNDICE DE FIGURAS

Figura 1 <i>Arquitectura de un sistema de detección de intrusos en redes WSN</i>	5
Figura 2 <i>Arquitectura de red de sensores inalámbricos</i>	12
Figura 3 <i>Estructura de un nodo de sensor</i>	14
Figura 4 <i>Topología en estrella</i>	16
Figura 5 <i>Topología en malla</i>	18
Figura 6 <i>Topología de clúster o árbol</i>	20
Figura 7 <i>Tipos de ataques</i>	30
Figura 8 <i>Ejemplos de topología en estrella y peer-to-peer</i>	75
Figura 9 <i>Distribución de los nodos de la red</i>	84
Figura 10 <i>Diagrama de secuencia de comunicación de los nodos sensores y el nodo coordinador</i>	85
Figura 11 <i>Distribución de los nodos de la red con la solución para la integración del IDS</i>	86
Figura 12 <i>Diagrama de secuencia de comunicación del nodo sensor, concentrador y coordinador</i>	87
Figura 13 <i>Distribución de los nodos de la red con la integración del nodo malicioso</i>	88

Figura 14 <i>Diagrama de secuencia de comunicación del nodo malicioso, nodo sensor, concentrador y coordinador</i>	89
Figura 15 <i>Diagrama del diseño de la red dentro del software</i>	90
Figura 16 <i>Integración del IDS dentro de la WSN</i>	92
Figura 17 <i>Diagrama de secuencia del proceso que realizaría el nodo malicioso hacia la red, el efecto que tiene el IDS, envío de alerta y registro de eventos</i>	93
Figura 18 <i>Diagrama del diseño de la red dentro del software</i>	94
Figura 19 <i>Topología de la red y la adición de la inteligencia artificial</i>	95
Figura 20 <i>Diagrama de flujo del funcionamiento del modelo Isolation Forest</i>	97
Figura 21 <i>Diagrama de secuencia donde se incluye la inteligencia artificial</i>	98
Figura 22 <i>Diagrama de flujo del funcionamiento completo del sistema</i>	100
Figura 23 <i>Creación de los nodos de la topología WSN</i>	102
Figura 24 <i>Configuración de los enlaces de los nodos sensores y modelo de propagación</i>	103
Figura 25 <i>Configuración de la generación de la lectura de temperatura y envío de la lectura de la simulación</i>	104
Figura 26 <i>Configuración de la réplica del tráfico de la red</i>	105
Figura 27 <i>Agregar el nodo IDS usando Snort</i>	106

Figura 28 <i>Edición del archivo general para el funcionamiento de Snort</i>	106
Figura 29 <i>Configuración del archivo de entrenamiento del modelo</i>	109
Figura 30 <i>Configuración del archivo de análisis de la captura de paquetes de la red con el modelo entrenado</i>	110
Figura 31 <i>Gráfica de la red de sensores inalámbricos</i>	112
Figura 32 <i>Comprobación de conectividad de la red</i>	113
Figura 33 <i>Salida de alertas de detección de Snort</i>	114
Figura 34 <i>Salida de alertas de detección de anomalías del modelo Isolation Forest</i>	115
Figura 35 <i>Ataque de inundación de mensajes ICMPv6</i>	116
Figura 36 <i>Ataque de inundación de redirecciones ICMPv6</i>	116
Figura 37 <i>Comprobación de la funcionalidad del sistema desarrollado para una red WSN</i>	118

ÍNDICE DE TABLAS

Tabla 1 <i>Algoritmos de detección</i>	54
Tabla 2 <i>Definición de los Stakeholders</i>	60
Tabla 3 <i>Definición de los requerimientos del sistema</i>	61
Tabla 4 <i>Requerimientos de arquitectura del sistema</i>	63
Tabla 5 <i>Verificación de los criterios para el sistema a implementar conjunto con los softwares de simulación que se analizaron</i>	65
Tabla 6 <i>Especificaciones de los requerimientos mínimos para el software</i>	67
Tabla 7 <i>Características principales para la identificación del ataque</i>	68
Tabla 8 <i>Algoritmos de detección para la identificación del modelo a implementar</i>	70
Tabla 9 <i>Descripción de los sistemas operativos con respecto a los requerimientos establecidos para la implementación del sistema</i>	72
Tabla 10 <i>Modelos de propagación que se pueden utilizar para redes inalámbricas</i>	78
Tabla 11 <i>Distancias que alcanzan los módulos Lora RN2903</i>	81
Tabla 12 <i>Ganancia de la antena definida según el entorno: interiores y exteriores</i>	83
Tabla 13 <i>Configuración de alertas para la detección de ataques</i>	107
Tabla 14 <i>Plan de pruebas</i>	111

Tabla 15 <i>Salida de alertas de la detección con el IDS</i>	114
Tabla 16 <i>Tráfico normal de la red</i>	119
Tabla 17 <i>Tráfico anómalo de la red</i>	119
Tabla 18 <i>Tráfico anómalo de la red</i>	120
Tabla 19 <i>Resultados finales del tráfico de la red</i>	120
Tabla 20 <i>Cantidades estimadas para la detección de los ataques en la red</i>	121
Tabla 21 <i>Resultados finales de la detección de los ataques en la red</i>	121
Tabla 22 <i>Análisis de referencias bibliográficas para la definición de la arquitectura del proyecto</i>	140
Tabla 23 <i>Análisis de referencias bibliográficas para la definición del ataque a implementar</i>	145
Tabla 24 <i>Análisis de referencias bibliográficas para la definición del algoritmo de detección del proyecto</i>	151

RESUMEN

Las WSNs usan nodos sensores que se comunican entre sí de manera inalámbrica, posibilitando la transmisión de información a través de la red. Este tipo de red es usada en diferentes aplicaciones como salud, agricultura, monitoreo ambiental y seguridad. Además de ofrecer flexibilidad y adaptación a diversos entornos mejorando la eficiencia y capacidad en diferentes escenarios. Sin embargo, debido a su naturaleza pueden ser susceptibles a ataques y amenazas de seguridad, destacando así la importancia de implementar medidas de protección.

El presente documento se centra en el diseño, implementación y evaluación de un Sistema de Detección de Intrusos Aplicando Inteligencia Artificial para la Detección de Ataques en una Red de Sensores Inalámbricos (WSN). La implementación de este sistema se lleva a cabo en un entorno simulado, con el fin de que se detecte y alerte a la red en caso de que se descubran vulnerabilidades específicas.

El desarrollo del sistema comprende varias etapas, comenzando por el análisis de las vulnerabilidades y amenazas que enfrentan las WSNs, análisis que permite que se establezcan lineamientos para la detección de intrusos, e incluso requerimientos para el IDS con la integración de IA que pueda ser capaz de aprender y acoplarse a patrones de comportamiento dinámico. Una vez se tienen implementadas las herramientas requeridas se realizan pruebas exhaustivas donde se comprueba el funcionamiento del sistema para detectar las intrusiones simuladas dentro del entorno de la red.

ABSTRACT

WSNs use sensor nodes that communicate with each other wirelessly, enabling the transmission of information through the network. This type of network is used in different applications such as health, agriculture, environmental monitoring and security. In addition to offering flexibility and adaptation to different environments improving efficiency and capacity in different scenarios. However, due to their nature they can be susceptible to attacks and security threats, thus highlighting the importance of implementing protection measures.

This paper focuses on the design, implementation and evaluation of an Intrusion Detection System Applying Artificial Intelligence for Attack Detection in a Wireless Sensor Network (WSN). The implementation of this system is carried out in a simulated environment, in order to detect and alert the network in case specific vulnerabilities are discovered.

The development of the system involves several stages, starting with the analysis of the vulnerabilities and threats faced by WSNs, analysis that allows the establishment of guidelines for intrusion detection, and even requirements for the IDS with the integration of AI that can be able to learn and adapt to dynamic behavior patterns. Once the required tools are implemented, exhaustive tests are carried out to check the system's performance in detecting simulated intrusions within the network environment.

CAPÍTULO I ANTECEDENTES

1.1. Problema

Los avances de las Redes de Sensores Inalámbricos (WSN) se utilizan en entornos donde se puede recolectar información gracias a los cambios tecnológicos presentes en la actualidad; las WSN permite la construcción de sistemas automatizados que conectan simultáneamente diferentes dispositivos y aparatos que realizan funciones específicas (Heredia Rivadeneria & Lucero Andrade, 2021). Por otro lado, se considera a este tipo de red muy adaptable, que puede organizarse para dar acceso a diferentes formas de recolección de información (Vázquez Rodas et al., 2021). Sin embargo, el despliegue de dichas redes presenta algunos inconvenientes como: limitación de recursos y procesamiento de datos, estableciendo que estos son procesos decisivos, además de promover eficiencia y seguridad dentro del sistema, siendo ámbitos que representan un desafío en temas de confiabilidad de los datos, por lo que la protección de la información es un área importante en ataques y amenazas con respecto a la seguridad (Gutiérrez Portela et al., 2021).

En cuanto a seguridad de la información o avances de mecanismos de defensa según la norma ISO/IEC 27001:2013 establece que los principios que se deben tener en cuenta son: la preservación de la confidencialidad, integridad y autenticidad (ISO/IEC, 2013). De acuerdo con lo mencionado, ahora se definen algunos de estos principios de seguridad y mecanismo de defensa como, por ejemplo: el protocolo DTLS fundamentado para sensores inalámbricos, que es propuesto en campos de confidencialidad e integridad, protegiendo la transmisión de extremo a extremo entre nodos. Por otro lado, los mecanismos de seguridad se fundan en factores como distribución de claves aleatorias, esquemas de cifrado, control de acceso, funciones hash y firmas digitales. Dentro del campo de disponibilidad, se proponen métodos como sistemas de detección de intrusos y

autenticación, evitando cierto tipo de ataques, en cuanto a la preservación de privacidad en redes WSN, se encuentran procesos como encriptación, clave simétrica y seudónimos criptográficos (Gutiérrez Portela et al., 2021).

Considerando la presencia de principios y mecanismos de seguridad para redes WSN, existen las amenazas y ataques que pueden ser empleados, por ejemplo: el ataque de inundación en el cual aparece en diversas capas de la arquitectura WSN en donde, si hay un nodo malicioso este enviará paquetes de datos de manera continua a un objetivo en concreto, acabando con los recursos que tenía disponibles aquel nodo. Dichos ataques se pueden definir como ataques de denegación de servicios (DoS), se dan dentro de las capas de aplicación, presentación, sesión, transporte, red, enlace de datos y física (capas que conforman el modelo OSI) (Obaid & Abeer, 2020), también está el ataque de fuga de información o captura de tráfico definido como aquel en el que se pretende entender el comportamiento del tráfico de la red y de los nodos, donde se exploran patrones de mensajes, longitud y/o duración permanente dentro del nodo central (Gutiérrez Portela et al., 2021), y finalmente el ataque Sybil donde un intruso puede producir diversas identidades en la capa de red y los paquetes a transmitirse contienen identidades falsas, los cuales pueden ser reformados de manera selectiva o descartados (Vasudeva & Sood, 2022).

La tecnología relacionada a las WSNs ha ido atrayendo mucho más la atención (Vlajic et al., 2011), debido a su mejoría en amplios métodos de alcance de aplicación y menor coste (Ahmed et al., 2015). Por lo tanto, el despliegue de dicho avance requiere la expansión de seguridad de este tipo de redes WSN, así como de todas las redes de comunicación alámbricas e inalámbricas en general. Además, hay otras características que garantizan la seguridad de la comunicación inalámbrica incluso cuando los recursos son limitados en comparación con las redes cableadas tradicionales. Por eso, acorde a lo detallado se funda la idea de un sistema de detección de intrusos

en una WSN, considerando que hay diversas herramientas, métodos o técnicas que contribuyan con el proceso de vulnerar las WSNs, es así como se establece un análisis de ataques para de esta manera conocer cómo actúan y establecer alertas o alarmas si existe algún tipo de actividad dentro de uno de los nodos o de la red, por lo que, un IDS puede ser mucho más productivo si se implica estrategias de inteligencia artificial (Kunal & Dua, 2019), trabajando con un cierto grupo de datos para entrenarlos y finalmente realizar pruebas para comprobar su correcto funcionamiento.

1.2. Objetivos

1.2.1. Objetivo General

Desarrollar un sistema de detección de intrusos aplicando inteligencia artificial para la detección de ataques en una red de sensores inalámbricos.

1.2.2. Objetivos Específicos

- Analizar las vulnerabilidades y amenazas en una WSN para tener una línea base para detección de intrusos.
- Establecer requerimientos para el diseño del sistema de detección de intrusos integrando la inteligencia artificial para la detección de alertas de seguridad.
- Implementar el IDS de intrusos en una red simulada para comprobar el funcionamiento individual de los componentes.
- Realizar pruebas de funcionamiento basada en la simulación de la WSN que evalúen el desempeño del sistema considerando algunos tipos de ataques.

1.3. Alcance

El proyecto se desarrollará en base a la metodología de investigación Agile, en la cual se representa una serie de fases o pasos que se detallan a continuación: fase 1 evaluación de procesos,

fase 2 sugerencia de mejora y optimización de procesos, fase 3 diseño de la aplicación, fase 4 construcción e implementación de la aplicación y fase 5 evaluación y monitoreo (Sadaf et al., 2017), empleadas con el cumplir el objetivo principal de desarrollar un sistema de detección de intrusos aplicando inteligencia artificial para la detección de ataques en una red de sensores inalámbricos.

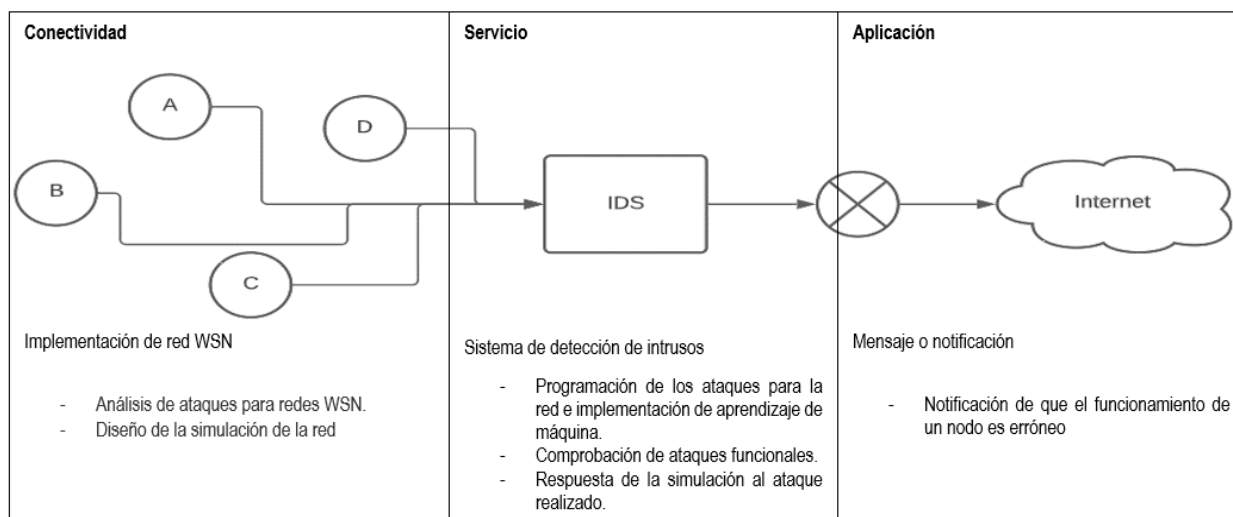
De acuerdo al primer objetivo específico planteado se establece que para lograr su cumplimiento total se desarrollará enmarcado en las fases 1 y 2 de la metodología, dado que para la fase 1 se establece la aplicación del sistema basado en la evaluación de los procedimientos que se van a llevar a cabo, centrándose en el análisis de los conceptos de ataques que son relevantes y potencialmente amenazantes para las WSNs, es decir, se espera que el análisis efectuado proporcione suficiente conocimiento para ser integrado dentro del sistema. A partir del análisis de los ataques y del diseño de la red, da comienzo con la fase 2 donde se ejecuta el proceso de desarrollo de un procedimiento para mejorar y optimizar la configuración resultante, aplicando conceptos y métodos de aprendizaje de máquina, una vez implementada la configuración se pueden examinar según las funciones existentes, en resumen, es necesario mejorar la funcionalidad y el rendimiento para poder desplegar un IDS en la red.

El segundo objetivo específico referente a requerimientos y diseño de sistema de detección de intrusos se ejecutará dentro de la fase 3, donde se espera que la red WSN diseñada debe ser manipulable dentro del simulador de redes, en este caso que permita la implantación de nodos y la aplicación de configuraciones para facilitar la introducción del IDS en la red, es decir, procedimientos de interconexión que garanticen el correcto funcionamiento del sistema y verificación de comportamiento de cada uno de los componentes de la red. A continuación, se observa cada uno de los procesos a ejecutar para el cumplimiento total del desarrollo del proyecto

de titulación y una base preliminar de una arquitectura de sistemas de detección de intrusos, esta información se encuentra detallada en la Figura 1.

Figura 1

Arquitectura de un sistema de detección de intrusos en redes WSN.



Nota. El gráfico detallado representa una arquitectura preliminar de un sistema de intrusos en redes WSN, además de las características que con lleva cada una de sus capas. Fuente: Autoría.

Para el cumplimiento del tercer objetivo específico se considera la fase 4, donde la implementación del sistema de detección de intrusos requiere, la producción de una comunicación adecuada entre la red y el sistema, procesos que se convertirán en interpretaciones para el tratamiento del despliegue del sistema total, además de detectar cualquier problema que pueda impedir que el IDS o por componentes individuales cumplan su objetivo.

Finalmente, el último objetivo específico se basa en la realización de pruebas de funcionamiento de la solución propuesta, proceso que se ejecutará dentro de la fase 5 de la metodología, en el cual, una vez realizados los ajustes necesarios, se realizan pruebas para observar cómo se comporta el IDS desplegado en la red. Las pruebas ejecutadas pueden producir resultados como el envío de

alertas de que se está efectuando un ataque en la red que podría ser atacada y vulnerada, o en su efecto que notifique acerca de nodos que tengan un comportamiento extraño dentro del entorno. Si no se obtienen los resultados esperados, es probable que se realicen cambios para resolver los problemas existentes hasta que el IDS en cuestión sea operativo.

1.4. Justificación

El tráfico en línea aumenta exponencialmente con el tiempo, lo que dificulta su gestión y exige nuevas tecnologías. Con el aumento de los flujos de datos, el valor de los datos para los usuarios es cada vez mayor y más importante, hasta el punto de que terceros los quieren. Esto se debe a que los datos pueden contener información vital para el funcionamiento normal dentro de su entorno (Zulfadhilah, 2017).

Las redes de sensores inalámbricos (WSN), concebidas y creadas en función de diversas aplicaciones reales, como la vigilancia del medio ambiente, el control de la salud, la automatización de procesos industriales, la vigilancia y el control de terremotos, han dado lugar a un rápido desarrollo de las tecnologías de comunicación y de sensores inalámbricos. Este tipo de red es atractiva porque es rentable y puede utilizarse en una amplia gama de situaciones críticas (Silnik et al., 2021).

Los principales objetivos de las WSN son la confidencialidad o privacidad, la integridad, la autenticación y la disponibilidad, además normalmente estas redes tienen que compartir el espectro de frecuencias con diversos servicios al igual que compenetrarse con protocolos similares o diferentes (O'Mahony et al., 2020), también se autoorganizan, se auto reparan y tienen una topología dinámica en entornos de saltos múltiples, lo que las convierte en vulnerables a ataques maliciosos. También son sensibles a muchos ataques de seguridad, y el grado de pérdida de

información varía en función del entorno físico y de los ataques no autorizados (Keerthika & Shanmugapriya, 2021).

Entre las tecnologías más recientes se encuentra la inteligencia artificial, en la que las máquinas imitan el comportamiento humano. El componente más importante para detectar ciberataques y actos maliciosos es el sistema de detección de intrusos (IDS). La inteligencia artificial desempeña un papel importante en la detección de intrusos y a menudo se considera la mejor manera de adaptar y desarrollar un IDS. Los algoritmos de inteligencia artificial atraen actualmente la atención como una nueva técnica computacional para la resolución de problemas en tiempo real (Kanimozhi & Jacob, 2019).

El presente proyecto intenta dar una solución para las amenazas y vulnerabilidades que pueden ser aplicables en una WSN, por lo que la implementación del IDS sumada la inteligencia artificial sirve en gran medida para contrarrestar dichos procesos debido a que, al diseñarlo ayuda con la detección temprana de anomalías ya sea dentro de la red o nodo que la conforma, mejorando así la manera de gestión de seguridad de la información, en otras palabras, el analizar los ataques admite que se implemente una alerta en los nodos que serán implicados en la topología, enviando así este mensaje de alerta. Representando una gran contribución con un administrador de red para que implemente medidas de acción inmediatas cuando su red se vea afectada.

CAPÍTULO II MARCO TEÓRICO

En este capítulo se aborda la revisión bibliográfica de los principales conceptos y definiciones para entender: qué es, cómo se configuran e implementa la seguridad de la información en WSNs. Por otro lado, también se conocerá sobre los sistemas de detección de intrusos, sus tipos y funciones existentes, y la relación entre las WSN y los IDS.

2.1. Red de Sensores Inalámbricos

Las Redes de Sensores Inalámbricas también conocidas como WSNs se hicieron muy populares desde el siglo XXI, estas se encuentran constituidas por un número de nodos conectados a una red, el hardware típico es primitivo, a prueba de manipulaciones, con grandes memorias, chips GPS, entre otros. La creación de una infraestructura de WSN no es muy complicada, un ejemplo de ellas puede ser: un conjunto de detectores que supervisa una zona específica y proporciona informes detallados sobre esta, a comparación de varios sistemas de red, las WSN tienen limitaciones inherentes de diseño y recursos (Suganya et al., 2019).

Considerando así a una WSN como una red autoorganizada fundamentada en datos, se encuentra formada por dispositivos accesibles, de potencia reducida y poca capacidad, usados en sistemas de comunicación inalámbrica de alcance limitado. El pequeño tamaño de los sensores y la naturaleza primitiva del hardware hacen que sean propensos a errores y fallos, concluyendo que tienen un rendimiento muy definido debido a su diminuta dimensión.

La naturaleza de la comunicación inalámbrica en las redes de sensores dificulta la protección de las comunicaciones frente a atacantes externos. El problema se agrava por el hecho de que los nodos no pueden realizar cálculos criptográficos complejos debido a su limitada potencia. Los

dispositivos de monitoreo de las WSN constan de microcontroladores, receptores y transmisores de radiofrecuencia, fuentes de alimentación y memorias (Suganya et al., 2019). Son aptos para ser aplicados en diversas áreas como por ejemplo vigilancia de condiciones ambientales, temperatura, humedad, agricultura, contenido del agua y salinidad de suelo, etc (Sun et al., 2010).

2.1.1. Características

Las WSN se usan para medir parámetros en entornos reales sin supervisión. Por lo tanto, es primordial considerar las particularidades de las WSNs para diseñar una red eficaz. Sun et al (2010) presentan una lista de las características principales que se describen a continuación:

Bajo Coste. Suelen utilizar cientos o miles de nodos sensores para medir el entorno físico, pero para reducir el precio total de toda la red, se procura que el costo individual de cada nodo sensor sea lo más económico posible.

Eficiencia Energética. En las WSNs, la energía se utiliza para diversos fines, como el cálculo, la comunicación y el almacenamiento. Comparativamente, el mayor consumo energético para la transmisión se da en un nodo sensor frente a otros tipos de sensores. Cuando el recurso se agota, quedan invalidados al no poder recargarse. Por lo tanto, el gasto del insumo debe tenerse en cuenta en la fase de diseño al desarrollar protocolos y algoritmos.

Potencia de Cálculo. Un nodo suele tener una capacidad de procesamiento limitada, se debe considerar en cuenta el coste y la energía.

Capacidad de Comunicación. Suelen comunicarse mediante ondas de radio a través de un canal inalámbrico. También sería capaz de transmitir datos a corta distancia en una banda de frecuencia estrecha y dinámica. La vía de transmisión puede funcionar en ambas direcciones o

sólo en una. Debido al entorno incontrolado y hostil, es difícil organizar un despliegue sin problemas. Por ello, el hardware y el software de comunicación deben desarrollarse teniendo en cuenta la fiabilidad, la seguridad y la flexibilidad.

Seguridad y Confidencialidad. Cada nodo sensor debe contar con medidas de protección adecuadas para evitar accesos no autorizados, ataques y cambios accidentales en la información contenida. Además, deben preverse otros mecanismos para proteger la privacidad.

Descubrimiento y Procesamiento Distribuido de los Sensores. Un gran número de nodos debe estar desplegado de forma uniforme o aleatoria. Cada nodo de la WSN puede recoger, clasificar, procesar, resumir y enviar datos al receptor. La detección distribuida garantiza así la estabilidad del sistema.

Topología de Red Dinámica. Generalmente son redes dinámicas. Los nodos sensores pueden fallar por falta de batería u otras condiciones, la comunicación puede interrumpirse, pero también pueden añadir nuevos dispositivos a la red y considerar que su estructura cambia con frecuencia. Por lo tanto, las WSNs deben incluir funciones de reconfiguración y autoorganización.

Autoorganización. Como los nodos sensores existen sin saberlo en un entorno incontrolado y hostil, deben autoorganizarse o cooperar para adaptarse al algoritmo distribuido y formar automáticamente una red.

Soporte para la Comunicación Multisalto. Contienen múltiples nodos sensores. Por lo tanto, una forma de comunicarse con un sumidero o una estación base es pedir apoyo a un intermedio a lo largo de la ruta. Para notificar con otro nodo receptor o terminal principal fuera de su propia radiofrecuencia, se requiere una transmisión multitrayecto a través de un punto de transición.

Orientación en el Uso. Son muy diferentes de las redes convencionales. Se orientan a la aplicación y abarcan los sectores militar, medioambiental y médico. Los nodos se encuentran dispuestos en orden aleatorio y ordenados según el tipo de aplicativo.

Funcionamiento Fiable. Ya que los sensores se despliegan en entornos amplios y a veces inhóspitos. Por lo tanto, los nodos deben ser resistentes a los fallos y las perturbaciones. Por ello, precisan ser capaces de autodiagnosticarse, autocalibrarse y autorrepararse.

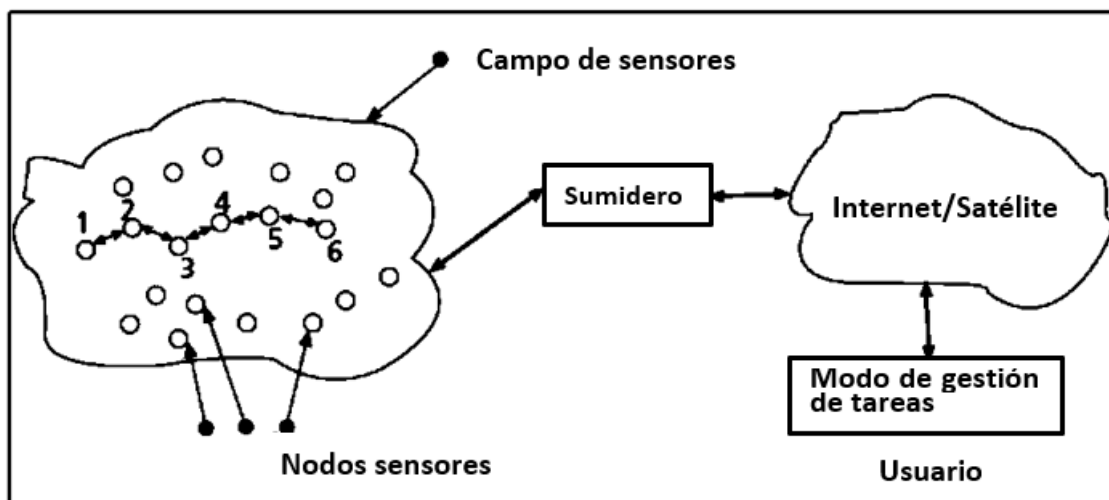
Pequeño Tamaño Físico. Los nodos sensores son generalmente pequeños y tienen un alcance limitado. Debido a su dimensión, su energía es limitada y, por tanto, su capacidad de comunicación es menor.

2.1.2. Arquitectura

Las WSN son reconocidas como una tecnología emergente que se está desarrollando y apoyando en diversas aplicaciones. Debido a sus tipos y medio, presentan una serie de retos para el desarrollo de los nodos. Es necesario entender como es el diseño, despliegue e implementación de una red de sensores y asimilar el mecanismos y herramientas flexibles que permitan un uso eficiente y adecuado. Por eso ahora se presenta una arquitectura típica de una red WSN (Yong-Min et al., 2009).

Figura 2

Arquitectura de red de sensores inalámbricos.



Nota. La arquitectura puede incluir la topología, la organización del nodo principal, otros nodos y la representación general de toda la red. La autonomía y la adaptabilidad son puntos importantes a tener en cuenta. Fuente: Adaptado de Yong-Min et al (2009).

Para cumplir con dicha arquitectura, se debería tener en cuenta los objetivos de diseño arquitectónico. Algunos objetivos importantes del diseño arquitectónico de las WSN son:

Determinar los Requisitos de la Aplicación de la WSN. En función de las necesidades de la aplicación a la que se destina, debe realizarse un análisis cuantitativo de la misma para facilitar el desarrollo de un diseño adecuado.

Identificar las Tendencias Tecnológicas Relevantes. Con el desarrollo de la microelectrónica, las tecnologías evolucionan exponencialmente. Es bien sabido que es un sistema heterogéneo y complejo. Con un sistema tan complejo, es importante tener en cuenta las limitaciones de coste y

diseño para encontrar la mejor solución de WSN que optimice el máximo rendimiento para las aplicaciones.

Diseño Optimizado. Los nodos sensores tienen recursos limitados. Por tanto, la red debe diseñarse de forma óptima para maximizar el uso de los nodos sensores y minimizar el consumo de recursos.

Métodos y Técnicas de Diseño. Se necesita una arquitectura basada en las tecnologías actuales y futuras. Las tecnologías actuales para alimentar y almacenar los componentes de los nodos de los sensores se consideran maduras. Las comunicaciones, los sensores y los actuadores inalámbricos de muy baja potencia se desarrollan casi a diario y todavía no son un fenómeno revolucionario. Por lo tanto, es importante determinar ya en la fase de diseño de la arquitectura qué tecnologías pueden utilizarse y cuáles deben desarrollarse.

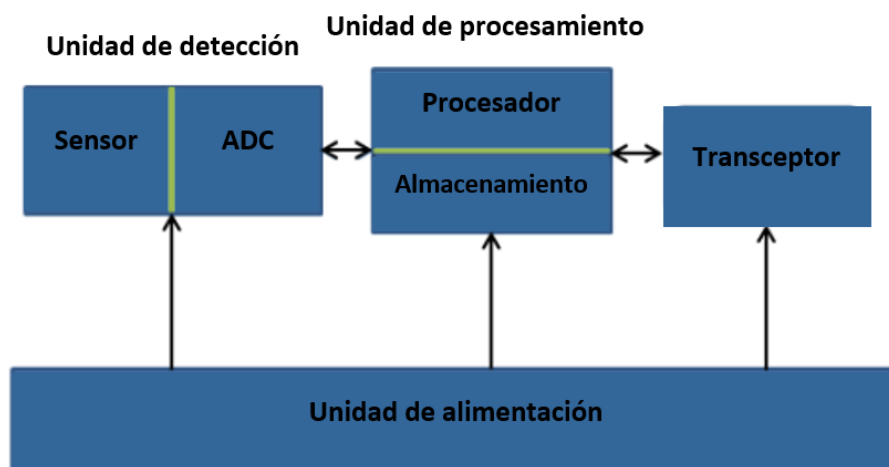
Análisis Cualitativo y Cuantitativo. Es necesario un análisis cualitativo y cuantitativo de las tecnologías, componentes y sensores existentes para diseñar una arquitectura de WSN eficiente y funcional.

La WSN es un sistema dinámico formado por diferentes nodos sensores. El entorno es heterogéneo en términos de hardware y software. El objetivo del diseño de los nodos sensores es reducir el coste, aumentar la flexibilidad y la tolerancia a los fallos (Ahmed et al., 2012).

Un nodo sensor consta de: un módulo sensor (sensor y convertidor analógico-digital), un módulo de procesamiento (procesador y memoria), un módulo de comunicación (transmisor) y una fuente de alimentación. En la Figura 3 se muestran las principales subunidades, seguidas de una breve descripción de cada una de ellas:

Figura 3

Estructura de un nodo de sensor.



Nota. El grafico detallado representa una estructura básica de cómo se encuentra formada un nodo de sensor. Fuente: Adaptado de Ahmed et al (2012).

Módulo de sensores. Consiste en diferentes tipos de sensores necesarios para medir diferentes eventos en el entorno físico. La elección de los sensores depende de su uso previsto. La señal de salida del sensor es una señal eléctrica analógica. Por lo tanto, se utiliza un convertidor analógico-digital (ADC) para convertir la señal en una señal digital para transmitirla al microcontrolador.

Módulo procesador. Consta de un procesador (microcontrolador) y una memoria principal (RAM) y contiene también el sistema operativo y el temporizador. La unidad procesadora se encarga de recoger datos de diversas fuentes y no de procesarlos y almacenarlos. El temporizador se utiliza para ejecutar una secuencia tras otra.

Módulo de comunicación. Unidad de transmisión y recepción compuesta por un emisor y un receptor. La comunicación se realiza a través de canales de comunicación que utilizan un protocolo

de red. En función de los requisitos y la idoneidad de la aplicación, se suele utilizar un método de comunicación adecuado, como la comunicación por radio, infrarrojos u óptica.

Alimentación. La tarea de la fuente de alimentación es suministrar energía al nodo sensor a bajo coste y en poco tiempo y monitorizar el entorno. La vida útil del sensor depende de la batería o del generador conectado al aparato. La forma de alimentar la batería es importante para su uso eficiente.

2.1.3. Topologías

Los nodos suelen estar distribuidos aleatoriamente porque su ubicación y conectividad determinan la topología de la red. En la práctica, se han desarrollado tres tipos básicos de topologías:

- Estrella
- Malla
- Clúster/árbol

Los nodos de estas topologías básicas pueden dividirse en dos categorías de áreas: Dispositivos totalmente funcionales - FFD (nodos de enrutamiento) y Dispositivos Funcionales Restringidos - RFD (nodos de terminación) (Ahmed et al., 2012).

El FFD tiene todas las funciones y puede desempeñar el papel de coordinador de red CN (Gateway). Los RFD pueden utilizarse como dispositivos usuales. Tienen menos memoria y recursos informáticos y consumen menos energía.

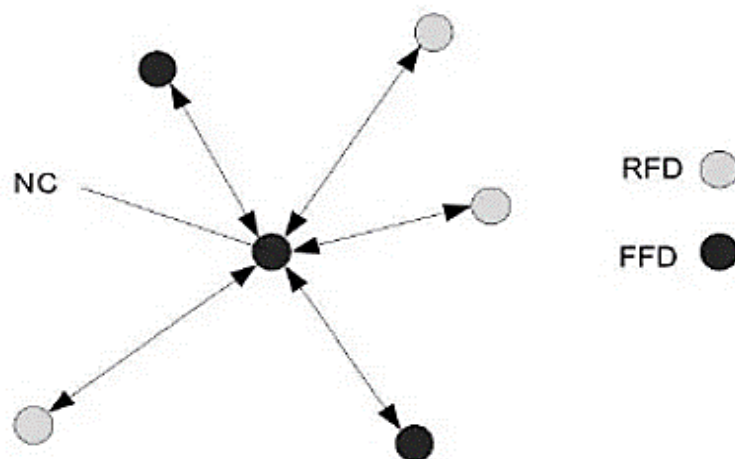
Topología en Estrella. Una topología en estrella es una red en forma de estrella con un núcleo central y muchos sistemas que están directamente conectados a este núcleo. Los sistemas de una

topología en estrella no están conectados entre sí, sino que envían mensajes al núcleo central, que a su vez envía mensajes a todos los demás sistemas o a un sistema de destino específico, dependiendo de la estructura de la red. Esta topología es adecuada para muchas redes pequeñas y evita muchos de los inconvenientes asociados a las topologías de bus o anillo. La topología en estrella tiene sus limitaciones, pero hay formas eficaces de sortearlas. De hecho, sólo se puede conectar un número limitado de sistemas a una única red en estrella antes de toparse con límites físicos, como la longitud de los cables o el número de puertos disponibles en el hardware utilizado para la red. La topología en estrella resuelve este problema, ya que puede ampliarse fácilmente a múltiples redes en estrella con un núcleo central en medio (Faircloth, 2014).

La Figura 4 representa una topología en estrella, donde los dispositivos pueden ser FFD o RFD. Actúan como terminales y se comunican directamente con los CN.

Figura 4

Topología en estrella.



Nota. La imagen contiene la estructura de una topología en estrella en una red WSN. Fuente: Adaptado de Kolev (2014).

Esta es la topología más sencilla. Esta designación se refiere a la disposición espacial de los nodos finales alrededor del nodo de coordinación. El CN es un dispositivo especial para la comunicación de alta velocidad, pero también puede ser un ordenador personal o portátil (PDA) cuya tarea es controlar las actividades de los terminales, intercambiar datos con ellos y transmitir los datos recogidos a otras redes.

Las topologías en estrella son sistemas unidireccionales, es decir, los datos se transmiten del emisor al receptor en una sola transmisión. Su estructura es de punto a multipunto. La principal ventaja de estas redes es que son las que menos energía consumen. El inconveniente es que el alcance de esta red es tan largo como el de los nodos finales.

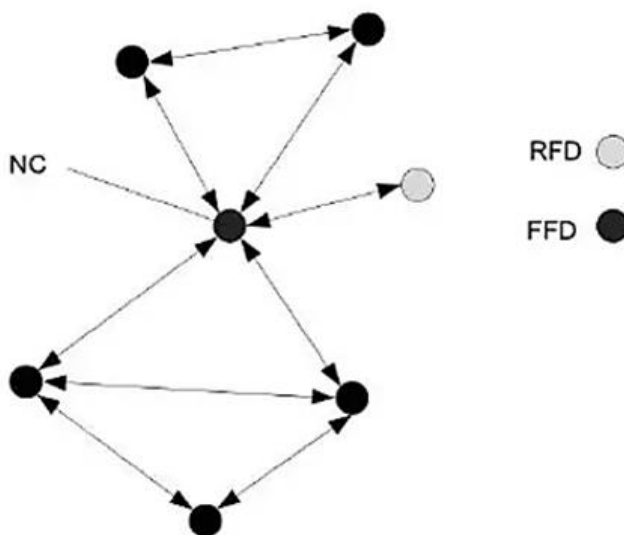
Además, tienen una baja fiabilidad: las interrupciones de la comunicación entre los nodos sensores y el coordinador de la red, por ejemplo, debidas a fallos, no se pueden compensar. Se recomienda que estas redes tengan un número máximo de 30 nodos y un alcance de 100 metros.

Topología en Malla. Esta topología está diseñada para proporcionar un nivel muy alto de redundancia, ya que cada sistema de la red está conectado a todos los demás sistemas de la red. A medida que se añaden más sistemas a la red, el número de conexiones entre ellos aumenta de forma natural. La alta redundancia de esta topología la hace atractiva para redes que necesitan estar disponibles en todo momento. Si falla un sistema o un cable de la topología de red, no puede provocar el fallo de la red porque siempre hay una ruta alternativa que puede utilizarse para comunicarse a través de la red. Para grandes redes que requieren una redundancia significativa o para redes locales muy pequeñas con requisitos de disponibilidad increíblemente altos, una topología de malla puede ser adecuada (Faircloth, 2014).

La Figura 5 representa la topología en malla, todos los dispositivos excepto los FFD más externos. En este caso, la comunicación con el CN también puede mantenerse mediante el reenvío de paquetes a múltiples nodos, lo que indica que estas redes son sistemas multinodos.

Figura 5

Topología en malla.



Nota. La imagen contiene la estructura de una topología en estrella en una red WSN. Fuente. Kolev (2014).

Además, son posibles múltiples caminos entre dos nodos porque el software de la red selecciona el camino más corto. Esto significa un menor consumo de energía, ya que se sabe que la potencia radiada necesaria es proporcional al cuadrado de la distancia. Para limitar los costes, el algoritmo de la WSN también garantiza una sincronización precisa del intercambio de datos: el nodo emisor envía un paquete de control que desencadena la acción del nodo receptor. Este último reacciona rápidamente, recibe los datos y se desconecta inmediatamente.

Si un nodo del router falla o se interrumpe una ruta importante, se encuentra automáticamente otro nodo, lo que hace que estas redes se autorreparen. Cuando se conectan y desconectan dispositivos y se añaden nodos móviles a la red (como en las redes GSM), la red se autorrepara cuando se produce un cambio de ruta. La autoconfiguración permite a la red detectar y conectar automáticamente cada nuevo nodo.

La estructura de la red se denomina punto a punto y peer-to-peer. A diferencia de las WSN con topología en estrella, en las que todo el control se concentra en un nodo CN, aquí el control se distribuye a todos los nodos, lo que convierte a este tipo de red en una red de control distribuido.

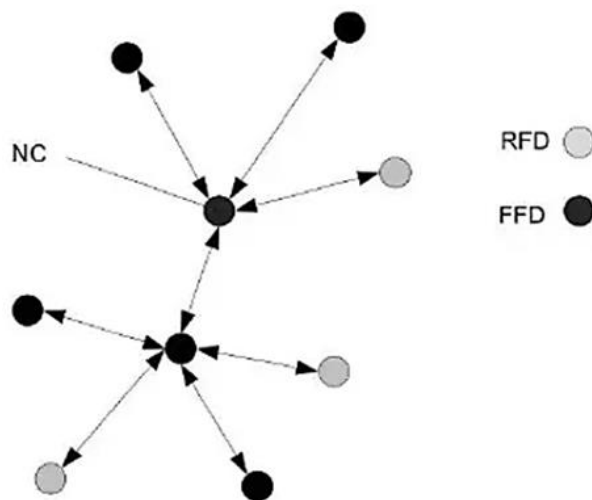
Además de la mejor conectividad de la topología de red, se puede cubrir un área mucho mayor en comparación con la topología en estrella.

Topología de Clúster/Árbol. La conexión en árbol cruzado es similar a una serie de redes en estrella interconectadas, salvo que no tiene nodo central. En su lugar, hay un nodo base, normalmente ocupado por un nodo o conmutador, desde el que se envían los demás nodos. Es un tipo de red en bus en la que el fallo de un nodo no provoca una interrupción de la comunicación. Se utiliza la misma ruta de comunicación. La topología en árbol puede considerarse una combinación de varias topologías en estrella. Tanto la topología en árbol como la topología en estrella son similares a la topología en bus, en la que el nodo de conexión funciona en modo broadcast y distribuye información a todas las estaciones, pero en esta topología las ramas se extienden desde la raíz (punto estrella) a tantas ramas como sea posible, dependiendo de las características del árbol (Leal Ávila Luis Daniel, 2019).

La topología de clúster se muestra en la Figura 6 se observa grupos de nodos que se comunican con un único FFD, definido como la cabeza de clúster.

Figura 6

Topología de clúster o árbol.



Nota. La ilustración a continuación contiene la estructura de una topología de clúster/árbol en una red WSN. Fuente. Kolev (2014).

La cabeza de clúster tiene más potencia, memoria y recursos informáticos. También puede reunir información adicional almacenando paquetes en la memoria y calculando los datos recogidos. A continuación, sólo transmite información aproximada y actualizada a la estación base de forma incremental (a través de otras cabeceras) sin sobrecarga.

En este tipo de red, cada grupo puede operar en una frecuencia diferente a la de sus vecinos, lo que evita colisiones y mejora la calidad de la comunicación en la red.

Cada clúster puede proporcionar mayor seguridad a la red utilizando diferentes claves para cifrar los datos. El enfoque de clúster mejora la escalabilidad y el rendimiento de la red.

En este caso también se observa una autoorganización, es decir, los nodos reconocen a sus vecinos y forman un clúster con un nodo central (cabeza). Esto ocurre sin la intervención de la dirección central de BS. Estos protocolos se utilizan en los casos en los que es necesario sustituir las cabezas de clúster para prolongar la vida del clúster y, por tanto, de toda la red.

Las aplicaciones de activación y suspensión se usan para recibir y transmitir información, para encender y apagar el transceptor. La transmisión sólo tiene lugar cuando cambian los parámetros comunicados por el sensor de información.

2.1.4. Aplicaciones

Las WSN pueden combinar diferentes tipos de sensores para controlar casi cualquier condición ambiental y ofrecen posibilidades para muchas aplicaciones, como las militares, sistemas de vigilancia de la salud, seguridad y vigilancia Haga clic o pulse aquí para escribir texto.(Dalal & Kukarni, 2021)Haga clic o pulse aquí para escribir texto..

A continuación, se amplía con más información de dichas aplicaciones de las WSNs según lo que menciona Dalal & Kukarni (2021):

Sistemas de Vigilancia de la Salud. Los sensores biocompatibles que se llevan en el cuerpo permiten recoger y utilizar grandes cantidades de datos en los ensayos clínicos, reduciendo el coste y las molestias asociadas a las visitas periódicas al médico. El sistema de vigilancia de la salud funciona de varias maneras. Por ejemplo, se utilizan algunas técnicas avanzadas para controlar la diabetes con una WSN o un sensor de presión arterial para aplicaciones biomédicas inalámbricas. Los pacientes pueden ser monitorizados y rastreados en casa, en las salas del hospital y en las unidades de cuidados intensivos en condiciones rutinarias o agudas.

Aplicaciones Militares. Al utilizar las WSN para el desarrollo de aplicaciones militares, es necesario reevaluar la importancia de las características mencionadas. Los sensores de imagen son útiles para una gran variedad de aplicaciones, como las militares, medioambientales, médicas, industriales y diversas aplicaciones de vigilancia. No es necesario que una aplicación incluya todas las características para ser eficaz. Sin embargo, las aplicaciones deben considerar aquellas que contribuyen a la productividad. Como se ha mencionado anteriormente, esta aplicación militar también tiene muchos aspectos. Las WSNs pueden ser utilizadas por los militares para diversos fines, como la vigilancia o el seguimiento de enemigos, la protección de fuerzas, los sistemas de navegación para vehículos militares, etc.

Seguridad y Vigilancia. Cada nodo sensor debe contar con las medidas de seguridad adecuadas para evitar accesos no autorizados, ataques y manipulaciones accidentales de la información contenida en el nodo sensor. En este tipo de aplicaciones, los nodos se instalan en lugares fijos en los que se controlan continuamente determinados parámetros para detectar posibles anomalías como incendios, gases tóxicos o incluso derrumbes de tejados mediante sensores micro sísmicos y de deformación de rocas. Si se tienen en cuenta las características básicas a la hora de diseñar la aplicación deseada, la aplicación resultante ofrecerá resultados más eficaces.

2.2. Seguridad de la Información

La seguridad de la información parece un concepto bastante simple y sencillo: se trata de la protección técnica de los sistemas y los datos frente a piratas informáticos, programas maliciosos y usos no deseados, así como del acceso a la información para reducir el riesgo para la empresa. Pero la seguridad de la información es algo más que la seguridad informática. No se trata sólo de cortafuegos, software antivirus y contraseñas seguras. En el mundo actual, la seguridad de la información plantea muchos riesgos para las empresas: el riesgo de violar las leyes de la

información, el riesgo de un grave daño a la reputación por la filtración de datos e información, el riesgo de fracaso empresarial debido a un fallo catastrófico del sistema de información y el riesgo de estar expuesto a actividades políticas continuas diseñadas para interrumpir las operaciones empresariales (Laybats & Tredinnick, 2016). Algunas de las definiciones de seguridad de la información abarcan varios aspectos de la gestión de la información y los datos: confidencialidad, integridad y disponibilidad.

Los componentes técnicos de la seguridad de la información son bien conocidos. Los cortafuegos supervisan, bloquean y filtran el tráfico de la red. Los programas antivirus, antispyware y antimalware analizan las aplicaciones y los datos en busca de contenido malicioso. Un fuerte cifrado protege los datos, el tráfico y las comunicaciones de las escuchas y las filtraciones accidentales.

El control de acceso, el control de versiones y los registros de auditoría ayudan a proteger la integridad de los sistemas informáticos. Estos elementos incluyen muros altos, cerraduras, puertas de seguridad y cortafuegos informáticos para impedir el libre flujo de información y garantizar el control. Sin embargo, es un error pensar que la seguridad de la información consiste únicamente en construir barreras, bloquear entradas y elegir las cerraduras más fuertes. La seguridad es algo que debe incorporarse a los sistemas y procesos de información desde el principio, no algo que se implanta a posteriori. Integrar la seguridad de la información en los procesos de gestión de la información significa comprender la naturaleza de las amenazas. Se tiende a sobrestimar las amenazas externas a la información y los datos, la amenaza de los piratas informáticos, los hackers políticos y los diversos tipos de malware y a subestimar las amenazas internas los empleados descontentos o negligentes.

Las amenazas a la seguridad de la información pueden dividirse en varios tipos:

- Los intrusos, los ataques de denegación de servicio, los programas maliciosos y espías, el espionaje industrial y las consecuencias de actos intencionados como el robo de datos, las filtraciones de datos o las violaciones deliberadas.
- Consecuencias imprevistas de actos intencionados, por ejemplo, la eliminación accidental o negligente de información, la divulgación accidental o negligente de información, la violación accidental de la confidencialidad, la fuga accidental de información.
- Consecuencias involuntarias de actos no intencionados, por ejemplo, pérdida accidental de datos, destrucción accidental de datos.

Los primeros son, en muchos sentidos, los más fáciles de predecir y los más fáciles de proteger. Los resultados esperados describen los tipos de actividad maliciosa y de software que reciben más atención: hacking, malware y robo de datos. Estos riesgos son relativamente fáciles de identificar: Son incógnitas conocidas en el mundo de la seguridad de la información, eventos que podemos predecir y para los que podemos prepararnos. Se puede diseñar políticas que prohíban a los empleados instalar software propietario y hacerles responsables si lo hacen; podemos formarles para que comprendan los peligros del malware y el spyware.

La seguridad de la información consiste en comprender y gestionar los riesgos, no en eliminar las amenazas. Si cada máquina informática en funcionamiento es también un ordenador en la red, no existe un sistema informático completamente seguro. Además de mantener la confidencialidad de la información, también es importante mantener la idoneidad de la propia información y de los procesos en los que se inserta, lo que inevitablemente introduce riesgos. Los sistemas más seguros

conlleven cierto riesgo para las empresas, y el equilibrio real entre la seguridad y la libre circulación de la información debe reevaluarse a diario.

En relación con los apartados anteriores, se ofrecen ahora definiciones más detalladas de los términos y elementos definidos en términos generales.

2.2.1. Modelo CIA

El modelo de la CIA define tres objetivos principales de la ciberseguridad (Nweke, 2017).

A continuación, se explica qué significan las siglas del modelo CIA según Romero Castro et al. (2018) *Introducción a la seguridad informática y al análisis de vulnerabilidad*.

2.2.2. Confidencialidad

La ciberseguridad exige la confidencialidad de los datos y la información. Se debe permitir o prohibir a determinadas personas, organizaciones o procesos la visualización de datos, archivos y objetos como nombres de usuario, combinaciones de contraseñas, historiales médicos, etc. La confidencialidad se refiere a restricción del acceso a la información a personas u organizaciones no autorizadas es decir negar el acceso a la información significa esencialmente que no puede llegar a manos de quienes queremos evitar que accedan a ella; para garantizar la confidencialidad se utilizan tres recursos principales:

- **Autenticación del usuario:** se utiliza para establecer que la persona que accede a la información es quien indica ser.
- **Gestión de derechos:** garantizar que los usuarios que tienen acceso al sistema sólo puedan procesar la información para la que están autorizados, y sólo de la manera en

que han sido autorizados, por ejemplo, gestionando los permisos de lectura o escritura de cada usuario.

- **Cifrado de la información:** también conocido como criptografía, impide que personas no autorizadas accedan a la información; para ello, la información se convierte de una forma legible a otra ilegible, lo que se aplica tanto a la información autorizada como a la no autorizada; sólo con un sistema de contraseñas se puede acceder a la información de forma legible, y esto se aplica tanto a la información recibida como a la almacenada.

Los principios de confidencialidad deben aplicarse a la protección de la información y a la protección de los datos de los que la organización es responsable. La información puede ser confidencial porque es de gran valor para la organización y porque puede estar sujeta a las leyes de protección de datos. Entre los ejemplos de violaciones de la confidencialidad se encuentran las filtraciones de información de bancos, grandes empresas y gobiernos que revelan partes de su actividad.

Los algoritmos criptográficos pueden dividirse en dos categorías principales: algoritmos criptográficos simétricos y asimétricos. Los algoritmos criptográficos simétricos tienen como principal desventaja que las partes implicadas deben tener acceso a una clave secreta común, un ejemplo de protocolo simétrico es AES-CCM, que define AES-CTR para el cifrado y AES-CBC para la generación de MAC. Por otro lado, el algoritmo criptográfico asimétrico utiliza un par de claves públicas y privadas para el cifrado y el descifrado. La principal desventaja de los protocolos asimétricos en el contexto del IoT es el alto consumo de energía en comparación con los protocolos simétricos. RSA y ECC (Elliptic Curve Cryptography) son ejemplos de primitivas criptográficas asimétricas (Mohammed Riyadh, 2016).

2.2.3. Integridad

La integridad significa mantener la exactitud y la integridad a la información a lo largo de su ciclo de vida, incluida la gestión y el análisis de los cambios en los datos o en la recopilación de datos. La filtración de información falsa puede ser tan perjudicial para una empresa como la pérdida de información. Si la manipulación de la información es lo suficientemente sutil, puede dar lugar a una cadena acumulativa de errores y posteriores decisiones equivocadas (Romero Castro et al., 2018). Para garantizar la fiabilidad de la información, hay que tener en cuenta los siguientes aspectos:

- Supervisión del tráfico de la red para identificar posibles fallos.
- Sistemas de control para aplicar la política de control para registrar quién, cuándo y qué información se ha comprobado.
- La implantación de sistemas de control de cambios, cosas sencillas como la comprobación del resumen de los registros almacenados en el sistema de control de cambios.
- Las copias de seguridad son otro recurso que permite restaurar la información a un estado anterior que no se puede evitar por manipulación o pérdida.

2.2.4. Disponibilidad

La disponibilidad garantiza que todas las medidas de ciberseguridad para el hardware, el software, las personas, los procesos y otras cosas permiten a los usuarios autorizados completar sus tareas. Esto significa que los usuarios autorizados deben acceder fácilmente a los recursos necesarios para trabajar, y se garantiza que el sistema es resistente y compensatorio en caso de incidente o desastre de ciberseguridad. Para que una información se considere fundamentalmente

segura, no es útil que sea accesible e inviolable sólo para el usuario; si el acceso es difícil o imposible, la información debe ser accesible para todos los que la necesitan para ser útil y valiosa (Mohammed Riyadh, 2016). Otro ejemplo de pérdida de accesibilidad es cuando una dirección de correo electrónico se utiliza para campañas de spam y luego se incluye en la lista negra para que los destinatarios legítimos no puedan recibir el correo electrónico. Esto requiere medidas de control como.

- Acuerdos de nivel de servicio (SLA).
- Balanceadores de carga de ancho de banda para reducir el impacto de los ataques DDoS.
- Copias de seguridad para recuperar la información perdida.
- Disponibilidad de recursos alternativos.

La información y los sistemas son seguros si el acceso a la información y a los recursos está restringido a las personas adecuadas, si se manipula o no intencionada de la información, esta puede detectarse y recuperarse, si se puede proporcionar un nivel de servicio aceptable y si se garantiza el acceso necesario a la información.

El uso de los sistemas de información requiere el establecimiento de políticas y procedimientos para el uso de los sistemas de información en el contexto de las amenazas potenciales, tales como:

- Desarrollar un conjunto de políticas y procedimientos.
- Definir las actividades que deben realizar las personas
- Definir el área de interés.

La seguridad de la información es la protección de la información de una organización, incluidos los datos, sistemas y recursos utilizados por la organización. Los términos ataque,

amenaza, vulnerabilidad y mitigación son conceptos clave en la seguridad de la información, algunos de los cuales se explican en las siguientes secciones:

2.3. Ataques, amenazas y vulnerabilidades a las WSNs

A medida que las redes evolucionan en función de su uso, aumenta el riesgo de amenazas, ataques y vulnerabilidades que pueden aprovecharse para interrumpir el flujo de información, manipular datos y obtener el control de nodos individuales. Por lo tanto, es importante comprender qué amenazas, ataques y vulnerabilidades pueden materializarse en las WSNs.

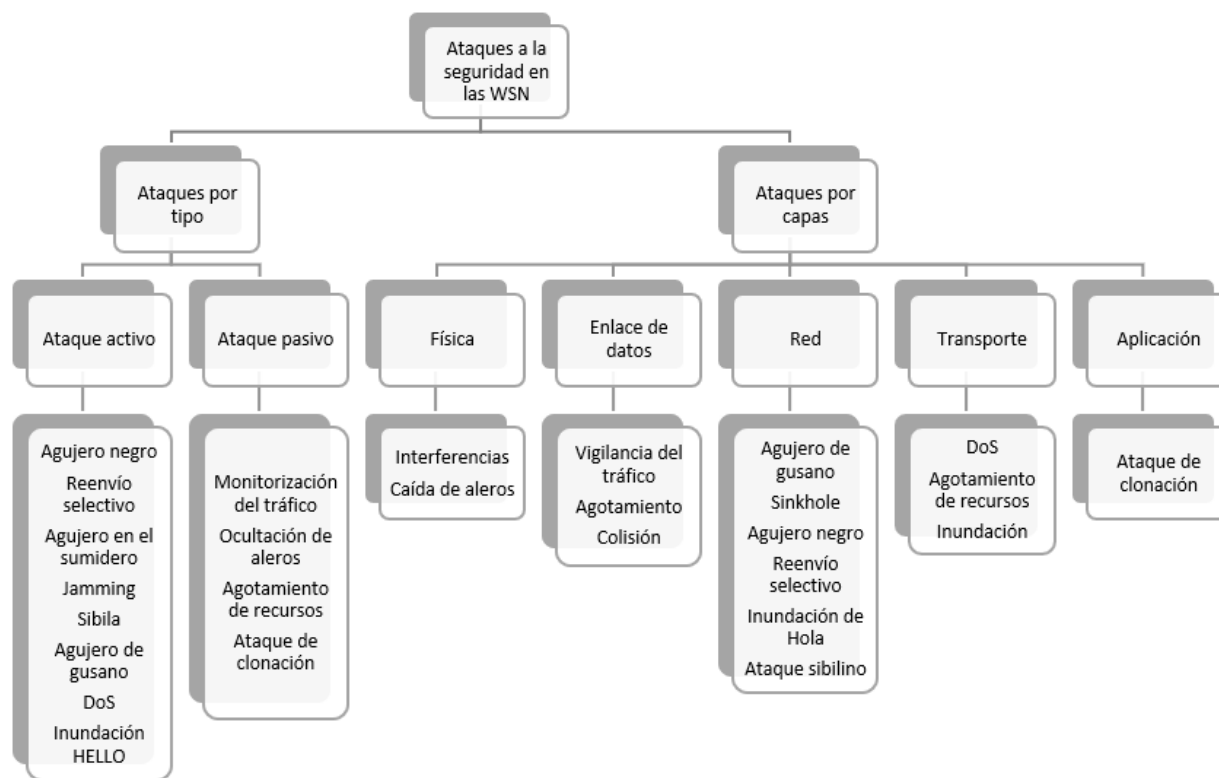
2.3.1. Ataques

Un ataque a una red de sensores inalámbricos (WSN) es un intento malintencionado de vulnerar la seguridad de la red o de sus dispositivos. Los ataques pueden producirse por diversos motivos, como el espionaje industrial, el robo de datos o la interrupción de la red.

La Figura 7 muestra ataques a las WSNs que se dividen en tres partes: orientados a los objetivos, orientados a la ejecución y orientados a las capas. Los ataques a la seguridad se dividen en dos partes: ataques estándar y ataques de la capa OSI (Kaur & Kaur Sandhu, 2021).

Figura 7

Tipos de ataques.



Nota. La ilustración muestra la clasificación de los ataques. Fuente. Kaur & Kaur Sandbu (2021).

Según la clasificación especificada dentro de la Figura 7 se observan los diversos ataques que se pueden implementar en una red WSN. Dentro de esta clasificación se determinan los ataques de forma activa y pasiva. Según lo que mencionado por Kaur & Kaur Sandbu (2021) se establece la definición de ellos y que tipo de ataques se encuentran en cada uno de ellos.

Ataques Activos. Los ataques activos son ataques en los que se modifican los datos (transmisión del nodo emisor al nodo receptor). Estos atacantes participan directamente en la transmisión de datos de un nodo a otro.

A continuación, se enumeran algunos ataques activos:

Ataque de Agujero Negro. En un ataque de agujero negro, un nodo sensor malicioso envía un mensaje RREP (Root Repeated Email Protocol) al nodo origen y al nodo emisor como el camino más corto hacia el nodo destino. El nodo fuente envía paquetes de datos al nodo malicioso. Finalmente, el nodo malicioso deja caer el paquete en lugar de enviar el paquete de datos completo al nodo de destino.

Variante de Enrutamiento de la Ruta. En un ataque de reenvío selectivo (SFA), el atacante se preocupa de descartar los paquetes de los dominios vecinos. El atacante suele reenviar datos irrelevantes, pero también puede negarse a reenviar datos importantes, como mensajes sobre las operaciones militares del adversario. Algunos diseños utilizan un enfoque multicopia, empleando múltiples caminos para transmitir los datos desde el nodo hasta el nodo de recepción. Muchos investigadores han utilizado múltiples copias de un paquete transmitido por múltiples caminos a una estación base para evitar la pérdida de información sensible.

Ataque de Inundación HELLO. Los nodos sensores suelen enviar un paquete "HELLO" para establecer una conexión entre otros nodos. Los atacantes activos envían mensajes "HELLO" a cada nodo vecino y generan tráfico de red. Estos mensajes provocan la congestión de la red y puede perderse información importante. En este caso, el nodo fuente no puede recibir ni intercambiar información.

Interferencias. Se trata de un conocido ataque a la comunicación entre nodos de una red. En un ataque de este tipo, se intercepta la información redundante transmitida por radiofrecuencias. La interferencia puede ser temporal, intermitente o permanente.

Ataque Sybil. Este ataque como el uso indebido de múltiples identificadores por parte de un atacante. En este ataque, el atacante crea un nodo duplicado en la red que es idéntico al nodo original. Cuando el emisor envía paquetes de datos, es difícil encontrar el nodo original y el nodo malicioso en la red.

Ataque Sink Hole. En este ataque, el nodo que almacena los datos es el nodo malicioso. El nodo envía los datos a la estación base para su posterior procesamiento. Cuando la fuente envía los datos al nodo de almacenamiento (nodo malicioso), la información puede perderse.

Ataque por Clonación. Los ataques de clonación son muy sencillos: un atacante quiere añadir un nodo sensor a una red ya en funcionamiento y utiliza el ID de red del nodo para hacerlo. Cuando el atacante sabe dónde está el nodo, puede sustituir fácilmente el real y añadirle el sensor clonado.

DDoS. Un ataque distribuido de denegación de servicio es un intento coordinado de usuarios engañosos para atacar un sistema con el fin de victimizarlo inundando el objetivo con un enorme tráfico de red en un corto espacio de tiempo. Esto se hace para sobrecargar el sistema objetivo para que sea incapaz de prestar el servicio adecuado, causando así la denegación de servicio a los usuarios legítimos. Básicamente, el objetivo está tan ocupado los paquetes de los intrusos que los paquetes de los clientes genuinos se llevan la peor parte, ya que la víctima apenas encuentra tiempo y recursos para atender sus peticiones. Esto impide a la víctima responder como se desea y puede resultar en el cierre de la red. Los ataques DDoS que tienen como objetivo inundar el tráfico para asaltar a la víctima se han vuelto muy temibles. En los últimos años han causado pérdidas

económicas y se han convertido en uno de los mayores riesgos para la seguridad de las WSN (Sahu et al., 2014).

Inundación. Un ataque de inundación es un ataque de denegación de servicio (DaS). El principal problema de un ataque de inundación es que el nodo atacante inunda toda la red. En un ataque de inundación, el atacante genera una solicitud de ruta y la envía sin comprobar si hay una ruta disponible en la tabla de enrutamiento. Cuando un nodo legítimo recibe una solicitud de ruta (RREQ), los nodos intermedios de la tabla de enrutamiento intentan centrarse en la ruta de destino y reenvían la solicitud a sus vecinos porque estos nodos tienen una ruta al destino. El objetivo principal del ataque es consumir energía, consumiendo una gran cantidad de batería y ancho de banda de la red. Esto acaba provocando diversos problemas de rendimiento de la red. El ataque de inundación provoca una degradación del rendimiento, el agotamiento de la batería y un uso ineficiente del ancho de banda. Un nodo malicioso puede modificar fácilmente el contenido de estos paquetes para llevar a cabo el ataque (HN Lakshmi et al., 2019).

Envenenamiento de la Tabla de Enrutamiento. En este caso, los nodos comprometidos de la red envían actualizaciones de enrutamiento falsas o modifican los paquetes de actualización de enrutamiento reales enviados por otro nodo honesto. El envenenamiento de la tabla de enrutamiento puede provocar un enrutamiento subóptimo, congestión en algunas partes de la red o incluso inutilizar algunas partes de la red (Sen, 2010).

Rushing. Un nodo atacante que recibe un paquete de solicitud de ruta de un nodo fuente lo reenvía rápidamente por la red antes de que otros nodos que hayan recibido el mismo paquete puedan responder. Los nodos que reciben paquetes de solicitud de ruta legítimos asumen que estos paquetes son copias de paquetes ya recibidos de un nodo competidor y, por tanto, los descartan.

Todas las rutas descubiertas por el nodo fuente incluyen al nodo rival como uno de los nodos intermedios. Por tanto, el nodo fuente no puede descubrir rutas seguras, es decir, rutas que no contengan un nodo resistente. Detectar este tipo de ataques en las WSN es extremadamente difícil (Sen, 2010).

Camuflaje. Un atacante puede apoderarse de un nodo sensor de una WSN y utilizarlo para camuflarse como un nodo normal de la red. El nodo camuflado puede entonces publicar información direccional falsa y atraer paquetes de otros nodos para su posterior transmisión. Cuando los paquetes llegan al nodo comprometido, este los reenvía a nodos estratégicos donde se puede analizar la confidencialidad de los paquetes. De lo anterior se desprende claramente que las WSN en todos los niveles de la pila de protocolos TCP/IP son vulnerables a múltiples ataques. Sin embargo, puede haber otros tipos de ataques que aún no han sido identificados. Proteger las WSN de todos estos ataques puede ser un reto importante (Sen, 2010).

2.3.2. Amenazas

Una amenaza se define como un acontecimiento o situación que podría poner en peligro el funcionamiento o la seguridad de una red y que no está necesariamente asociado a un ataque directo. Las redes de sensores inalámbricos se utilizan en muchas aplicaciones, como la vigilancia ambiental, la monitorización médica, la seguridad del hogar y la automatización industrial. Sin embargo, como cualquier red de computadoras, las WSN también están sujetas a diversas amenazas.

En los siguientes apartados se describen algunas de las amenazas identificadas por (MAGERIT, 2012):

Desastres Naturales. Hay sucesos que pueden ocurrir en la red sin intervención humana, como rayos, tormentas eléctricas, terremotos, huracanes, corrimientos de tierra y aludes de lodo. Los sucesos que pueden ocurrir sin intervención humana directa o indirecta son naturales o accidentales. Dentro de esta clasificación se encuentra el fuego y daños por agua.

- **Fuego:** Es una amenaza para los activos del sistema por la posibilidad de que sean destruidos por las llamas.
- **Daños por agua:** La amenaza de inundación significa que el agua puede destruir los activos del sistema.
- **De origen industrial.** Estos sucesos pueden ocurrir involuntariamente y son el resultado de la acción humana. Estas amenazas pueden producirse de forma involuntaria o intencionada. Se detallan a continuación algunas de ellas: **Desastres industriales:** Otros tipos de sucesos dañinos que pueden desencadenar las actividades humanas son las explosiones, los corrimientos de tierras, los accidentes de tráfico, la contaminación química, las subidas de tensión y las fluctuaciones de voltaje en la red eléctrica. Estos fenómenos adversos pueden provocarse tanto por el entorno como por acciones humanas, accidentales como intencionadas.
- **Contaminación mecánica:** Las vibraciones, el polvo y la suciedad pueden deberse a factores ambientales no intencionados, pero las acciones humanas, también pueden ser fuente de estos riesgos para los equipos o sistemas implicados.
- **Contaminación electromagnética:** La vibración, el polvo y la suciedad son algunos de los factores que pueden afectar al rendimiento del sistema. También se encuentran las interferencias de radio, los campos magnéticos y la radiación ultravioleta factores que pueden afectar al funcionamiento del sistema y emanar del entorno.

- **Avería de origen físico o lógico:** Las interferencias en la red pueden deberse a problemas de hardware y software. Estas perturbaciones pueden deberse a fallos incorporados al sistema o producirse durante su funcionamiento. En algunos sistemas puede ser difícil determinar si el fallo es de origen físico o lógico, aunque esta distinción suele ser irrelevante para las consecuencias. Los fallos pueden deberse a factores externos, como el entorno, o a acciones humanas, ya sean accidentales o intencionadas.
- **Condiciones inadecuadas de temperatura o humedad:** Las instalaciones inadecuadas pueden exponer los equipos a condiciones extremas como altas temperaturas, bajas temperaturas o alta humedad, que pueden ser causadas por accidentes medioambientales o por acciones humanas intencionadas o no.
- **Fallo de servicios de comunicaciones:** La pérdida de la capacidad de transmisión de datos entre instalaciones puede deberse a la destrucción física de los soportes o al cierre de las centrales. Puede estar causada por accidentes medioambientales o por acciones humanas accidentales o deliberadas cuando las centrales son incapaces de gestionar el tráfico existente.
- **Emanaciones electromagnéticas:** La amenaza consiste en que los datos internos se transmitan a terceros por radio, lo que convierte al emisor en víctima pasiva de un ataque. Esta amenaza suele denominarse "ataque TEMPEST". La amenaza puede ser involuntaria o el resultado de una acción humana accidental e intencionada.

Errores y Fallos No Intencionados. Los errores involuntarios causados por el ser humano son de naturaleza similar a los ataques intencionados, pero difieren en cuanto al objetivo que el autor intenta alcanzar. Estos errores involuntarios son errores humanos en el uso de servicios, datos y

otras cosas. Son el resultado de una acción humana no intencionada. La lista de estos errores no es secuencial, sino que corresponde a ataques intencionados.

Ataques Malintencionados. Son errores causados intencionadamente por el ser humano y pueden distinguirse de otros errores no intencionados, que pueden ser de naturaleza similar a los ataques intencionados, pero difieren en la intención de la persona responsable. Estos errores son el resultado de la acción humana.

2.3.3. Vulnerabilidades

Las vulnerabilidades del sistema son puntos débiles en el software o hardware de un servidor o cliente que un atacante puede explotar para obtener acceso o interrumpir la red, también se considera como una condición, debilidad o falta de procedimientos de seguridad o controles técnicos, físicos o de otro tipo que pueden ser explotados por una amenaza (Kizza, 2017).

Las WSN tienen varias vulnerabilidades que pueden ser explotadas por los atacantes. podemos profundizar en algunas de las principales vulnerabilidades que enfrentan estas redes:

Falta de Autenticación. La autenticación es el proceso que verifica la identidad de los nodos en una WSN. Sin un mecanismo de autenticación adecuado, los nodos pueden ser vulnerables a los ataques. Los atacantes pueden hacerse pasar por nodos legítimos y enviar información falsa a la red, lo que puede conducir a una mala toma de decisiones e incluso al fallo de la red. Por lo tanto, para evitar este tipo de ataques, deben implementarse mecanismos de autenticación fuertes, como el uso de certificados digitales o contraseñas seguras (Obaidat & Misra, 2014).

Falta de Cifrado. Un cifrado insuficiente de los datos transmitidos a través de las WSN puede permitir a los atacantes interceptar y leer la información. Esto puede ser especialmente peligroso

si los datos transmitidos contienen información personal o sensible, como datos médicos o financieros. Por ello, para proteger los datos transmitidos por la red debe utilizarse un cifrado potente, como el cifrado AES (Chowdhury & Fazlul Kader, 2013).

Topología de Red Débil. La topología de la red es un factor importante para la seguridad de las WSN. Si los nodos están en lugares donde pueden atacar fácilmente, la red puede ser vulnerable a ataques físicos. Por ejemplo, los nodos situados fuera de un edificio pueden ser más vulnerables a los ataques que los nodos situados dentro del edificio. Por lo tanto, hay que tener en cuenta la ubicación de los nodos y la topología de la red para minimizar el riesgo de ataques físicos (Kumar et al., 2011).

2.4. Mitigaciones de Ataques a WSNs

Para mitigar los riesgos en las redes de sensores inalámbricos (WSN), se pueden implementar una serie de medidas de seguridad y buenas prácticas, tales como:

2.4.1. SET: Detección de Clones de Nodos

Este método pertenece a la categoría de métodos basados en estaciones base. El método se basa en un enfoque basado en la red. En SET, la red se divide aleatoriamente en ciertos subconjuntos. Cada subconjunto tiene un líder y los miembros de la red se sitúan a un paso del líder del subconjunto. Se determinan aleatoriamente múltiples raíces para formar múltiples subárboles, y cada subárbol es un nodo de un subárbol. Cada líder de subárbol recopila información sobre sus miembros y la transmite a la raíz del subárbol. En la raíz de cada subárbol se realiza una operación de cruce para identificar los nodos duplicados. Una intersección vacía de todos los subárboles de un subárbol significa que no hay nodos clonados en ese subárbol. En el último paso, cada raíz envía su mensaje a la estación base. La estación base detecta los nodos clonados calculando la

intersección de los dos subárboles resultantes. SET detecta los nodos clonados enviando información sobre los nodos desde el líder del subárbol al nodo raíz de un subárbol generado aleatoriamente y, a continuación, a la estación base (Khan et al., 2013).

2.4.2. Algoritmo para la Detección de Ataque de Sumidero

Los ataques de sumidero pueden ser peligrosos para las WSN, ya que suelen funcionar con baterías largas y recursos limitados. Si un ataque "sinkhole" tiene éxito, puede consumir rápidamente la energía de un nodo vecino, acortando la duración de la batería y provocando cortes en la red. Las técnicas de mitigación y detección, como la autenticación entre pares, el cifrado de datos y la supervisión de la red, se utilizan para limitar el número de estos ataques y permitir una detección y respuesta rápidas ante actividades sospechosas o maliciosas en la red.

Dentro de un estudio se establece el desarrollo de un algoritmo sencillo para detectar los ataques sumidero. En su enfoque, se encuentra una estación base que recopila información sobre los flujos de red utilizando un enfoque distribuido, para luego, un algoritmo de identificación analiza los datos recopilados para detectar el sumidero. Su trabajo también considera el caso de una red con múltiples atacantes secretos (Shafiei et al., 2014).

2.4.3. Detección de Ataques DoS

Actualmente, para combatir los ataques DoS se utiliza un sistema de detección de intrusiones (IDS) basado en firmas. El proceso de detección de intrusiones en IDS consta de tres fases: la fase de recopilación de información, la fase de toma de decisiones y la fase de detección de intrusiones. El método utilizado para la detección consiste en comprobar el límite máximo de peticiones de enrutamiento (RREQ_RATELIMIT), que se establece en al menos 10 paquetes de petición de enrutamiento (RREQ) por segundo según el RFC 3561. La solución propuesta solo puede detectar

el ataque o no. Por lo tanto, se necesitan medidas adicionales para detenerlo y minimizar los daños causados por él.

La solución propuesta dentro del artículo “Mitigation and Detection Strategy of DoS Attack on Wireless Sensor Network Using Blocking Approach and Intrusion Detection System” donde menciona que la detección y mitigación de los ataques DoS en WSNs adoptando el esquema de cálculo RREQ_RATELIMIT y utilizando un IDS basado en firmas para la detección y añadiendo una función de bloqueo para mitigar el ataque agotando la energía del atacante. Cuando se produce un ataque DoS con peticiones RREQ que superan el RREQ_RATELIMIT, la WSN deja caer paquetes del nodo atacante para mitigar o incluso detener el ataque, ya que la energía del nodo atacante se consume por la transmisión continua (Kurniawan & Yazid, 2020).

2.5. Sistema de Detección de Intrusos (IDS)

IDS es un producto que aumenta la seguridad de la red y protege los datos de una empresa. Los IDS ayudan a los administradores de red a detectar actividades malintencionadas en la red y les alertan para que protejan sus datos tomando las medidas adecuadas contra esos ataques. Un intruso es un acceso no autorizado o un uso malintencionado de los recursos de información. Un intruso o usuario malintencionado es una entidad real que intenta encontrar la manera de obtener acceso no autorizado a la información, causar daños o realizar otras actividades maliciosas. Un sistema de detección de intrusos es un sistema de seguridad tipo cortafuegos. El cortafuegos protege a la organización de ataques maliciosos desde Internet, y el IDS detecta cuando alguien intenta traspasar el cortafuegos, cuando alguien ha conseguido saltarse la protección del cortafuegos y acceder al sistema de una organización, y alerta al administrador del sistema de la actividad no deseada en el cortafuegos (Tiwari Mahit et al., 2017).

En resumen, un sistema de detección de intrusos (IDS) es un sistema de seguridad que supervisa el tráfico de la red y del sistema informático y analiza este tráfico para detectar posibles ataques hostiles y abusos del sistema desde el exterior o ataques dentro de la organización.

2.5.1. Tipos

Según lo establecido por Tiwari Mahit et al. (2017) los IDS se pueden clasificar en dos categorías principales: IDS basados en red y IDS basados en host:

Sistema de Detección y Prevención de Intrusiones en Red (NIDS). Un sistema de detección y prevención de intrusiones en red (NIDS) reside en un ordenador o dispositivo conectado a un segmento de red de una organización y supervisa el tráfico de red en ese segmento de red para detectar ataques en curso. Se utilizan varios algoritmos hash, como MD5, para proteger los archivos en la red. Cuando se producen situaciones en las que el IDS de red debe ser consciente de un ataque, responde enviando alertas a los administradores. El NIDS busca patrones de ataque en el tráfico de red, como grandes colecciones de objetos relacionados de un determinado tipo, lo que puede indicar un ataque de denegación de servicio, o busca una serie de paquetes relacionados intercambiados en un determinado patrón, lo que puede indicar un escaneo de puertos. Un NIDS se instala en un lugar específico de una red (por ejemplo, en un router) para supervisar el tráfico que entra y sale de un segmento de red específico y puede utilizarse para supervisar ordenadores concretos de un segmento de red o para supervisar todo el tráfico entre sistemas de una red entera.

Sistema de Detección de Intrusos Basado en Host. Un sistema de detección de intrusos basado en host (HIDS) se instala en un ordenador o servidor específico, denominado host, y supervisa únicamente las actividades de ese sistema. Los sistemas de detección de intrusiones basados en host se dividen en dos categorías: métodos basados en firmas (detecto de usos

indebidos) y métodos basados en anomalías. Los HIDS monitorizan el estado de los archivos críticos del sistema y detectan cuando un atacante crea, modifica o borra los archivos monitorizados. Los HIDS activan una alarma cuando se produce alguno de los siguientes cambios: se modifican los atributos de los archivos, se crean nuevos archivos o se eliminan archivos existentes. La principal diferencia entre NIDS y HIDS es que NIDS puede acceder a la información cifrada mientras se transmite por la red.

- **IDS basado en firmas**

Un sistema de detección de intrusos (IDS) basado en firmas es un tipo de sistema de detección de intrusos que funciona comparando el tráfico de red o los archivos del sistema con una base de datos de firmas de amenazas conocidas o patrones de comportamiento malicioso.

Estas firmas pueden identificarse como patrones específicos de bytes o scripts característicos de ciertos tipos de ataques o malware. Si el IDS encuentra una coincidencia entre el tráfico de red o el archivo analizado y una firma de la base de datos, se activa una alerta que indica la presencia de una amenaza potencial (Anjum et al., 2003).

Los IDS basados en firmas son un método ampliamente utilizado y eficaz para detectar y defenderse contra amenazas conocidas, pero pueden ser menos eficaces contra nuevas amenazas o variantes de amenazas conocidas. Por lo tanto, es importante complementar este enfoque con otros métodos de detección, como los IDS basados en anomalías que buscan comportamientos inusuales en el tráfico de red o en los sistemas.

- **IDS basado en anomalías**

Un sistema de detección de intrusos (IDS) basado en anomalías es un sistema de detección de intrusos que utiliza el aprendizaje automático para detectar patrones inusuales en el tráfico de red en comparación con el tráfico de red normal.

En lugar de basarse en una base de datos predefinida de firmas de ataque conocidas (como ocurre con los IDS basados en firmas), los IDS basados en anomalías utilizan una "línea de base" de comportamiento normal para detectar amenazas potenciales (Jyothsna V et al., 2011).

El sistema examina y supervisa el comportamiento de la red a lo largo del tiempo para detectar patrones inusuales en el tráfico de la red. Si el sistema detecta patrones que se desvían significativamente de la línea de base, activa una alerta por un posible ataque de piratería informática.

Aunque los IDS basados en anomalías son una herramienta valiosa para detectar ataques sofisticados que no son rivales para los métodos de detección de intrusos más sencillos, pueden producir un gran número de informes de falsos positivos si no se configuran correctamente. Por lo tanto, es importante configurar el sistema para reducir el número de falsos positivos y aumentar la eficacia de la detección.

2.5.2. Componentes Funcionales Fundamentales

Muchos IDS pueden describirse mediante tres elementos funcionales principales según lo establece en Bace & Mell (2015):

Fuentes de Información: las diversas fuentes de información de eventos que pueden utilizarse para determinar si se ha producido una violación de la seguridad. Estas fuentes pueden proceder

de distintos niveles del sistema, pero suelen incluir la supervisión de la red, el host y las aplicaciones.

Análisis: un sistema de detección de intrusos que organiza y comprende los eventos de las fuentes de información y determina cuándo estos eventos indican que un ataque es inminente o que ya se ha producido. Los enfoques de análisis más comunes son la Detección de Explotación y la Detección de Anomalías.

Respuesta: es una serie de acciones que el sistema lleva a cabo cuando se detecta un intruso. Suelen dividirse en acciones activas y pasivas, donde acciones activas significan que el sistema reacciona automáticamente, mientras que acciones pasivas significan que los resultados del IDS se comunican a una persona que debe tomar medidas basándose en estos informes (Bace & Mell, 2015).

2.5.3. *Softwares*

Existen muchos softwares para el desarrollo de un IDS (Intrusion Detection System). Algunos de ellos son:

Snort. Snort fue publicado en 1998 por Martin Roesch y fue descrito originalmente como una tecnología de detección de intrusos "ligera". Un IDS ligero debe ser independiente de la plataforma, tener poco impacto en el sistema y ser fácilmente configurable para un administrador de sistemas que necesite implantar una solución de seguridad específica en poco tiempo. Snort llenaba entonces un gran vacío en los sistemas de seguridad de red. Era una aplicación sencilla que podía monitorizar pequeñas redes TCP/IP y detectar una amplia gama de tráfico sospechoso y ataques conocidos. También podía proporcionar a los administradores de sistemas información suficiente para decidir sobre actividades sospechosas. Snort también era capaz de cerrar

rápidamente brechas en el entorno de seguridad de la red, ya que, si un sistema propietario detectaba un ataque, tardaba mucho tiempo en publicar una nueva base de datos de firmas para eliminar el ataque, mientras que con Snort se podía actuar más rápido.

Snort es esencialmente un escáner de paquetes de red que actúa como sistema de detección de intrusiones y se basa en la biblioteca libpcap, una interfaz de registro de paquetes desarrollada como parte de la aplicación tcpdump. Esta biblioteca permite a los desarrolladores recibir y procesar paquetes de la capa de enlace. La principal característica que diferencia a Snort de tcpdump es que controla la carga útil de los paquetes. Esto permite a Snort detectar una amplia gama de actividades maliciosas, como desbordamientos de búfer, vulnerabilidades CGI y cualquier otra cosa que pueda encontrarse en la carga útil.

Snort está reconocido como el estándar de facto en protección y detección de intrusiones, con más de 4 millones de descargas y casi 400.000 usuarios registrados, lo que lo convierte en la tecnología de protección contra intrusiones más extendida del mundo. Utiliza un lenguaje de reglas flexible para la detección de movimientos en tiempo real y un motor de detección basado en una arquitectura modular.

La fuerza y difusión de Snort se debe en gran medida a la influencia y difusión de la comunidad de usuarios de Snort, formada por un gran número de desarrolladores que prueban y publican resultados y opiniones sobre la funcionalidad y el conjunto de reglas de Snort. Como Eric Raymond argumentó en su artículo y demostró posteriormente en el desarrollo de GNU/Linux, los errores encontrados en la comunidad de código abierto se corrigen con mayor rapidez y eficacia que en un entorno de desarrollo propietario (Haro Bermejo Francisco, 2015).

Dado que Snort es una aplicación de código abierto, tiene la ventaja de ser un sistema configurable que puede adaptarse a necesidades específicas. Esta es una de las razones por las que las grandes organizaciones, como las agencias gubernamentales y el ejército, eligen Snort para desarrollar sus propios sistemas de detección de intrusiones en lugar de utilizar aplicaciones propietarias que, en muchos casos, no pueden alcanzar el mismo rendimiento y las mismas capacidades que Snort.

Bro. Fue diseñado y desarrollado por Vern Paxson en el Internet Center for Internet Research (ICIR) del ICSI de Berkeley. El proyecto comenzó en 1995 en el Lawrence Berkeley National Laboratory (LBL), y Bro se ha desarrollado activamente desde entonces. Se publicó por primera vez en 1998. A Bro se sumaron importantes aportaciones de investigadores de la Universidad de California en Berkeley y de la Universidad de Princeton en Nueva Jersey. Mientras lo desplegaban en un gran entorno de investigación en el Centro Leibniz de Múnich a través de una conexión a Internet de 622 Mbps, se encontraron con problemas que a menudo se subestiman cuando se utilizan NIDS en redes más pequeñas.

El principal objetivo del proyecto Bro es desacoplar el mecanismo de la política. Aunque Bro implementa una gestión de estados y un análisis de protocolos muy sofisticados, es inherentemente independiente de la política. La actividad de la red se abstrae en eventos que se envían desde el núcleo de Bro a una capa superior: la capa de políticas. En esta capa, el administrador define las restricciones del entorno escribiendo escenarios personalizados en un potente lenguaje de scripting. Este diseño no hace que Bro sea inmune a anomalías o exploits (Sommer, 2003).

Por el contrario, se soportan ambos enfoques, y los scripts de políticas por defecto que acompañan a Bro utilizan ejemplos tanto de anomalías como de exploits. El comparador de firmas

incorporado proporciona un superconjunto de funciones Snort e incluso puede utilizar firmas Snort usando un convertidor.

Suricata. Es un sistema de detección de intrusos (IDS), un componente capaz de analizar el tráfico que entra y sale de la red en la que está instalado.

El propósito de instalar un IDS en una red (normalmente local) es monitorizar el tráfico para detectar actividades sospechosas y/o maliciosas contra el host, siempre que pueda monitorizar tanto el tráfico en la red como el tráfico que entra y sale de la red.

También realiza las funciones adicionales del Sistema de Prevención de Intrusiones (IPS) y del Mecanismo de Vigilancia de la Red (NSM) para prevenir intentos de intrusión o actividades de red potencialmente peligrosas para la seguridad de los hosts de la red. Esta funcionalidad es posible cuando el dispositivo IPS (denominado sensor o agente en el caso de los IDPS basados en host) está instalado en modo inline, lo que significa que el tráfico debe pasar por el sensor y ser analizado antes de llegar a la red interna. Este servidor se denomina servidor de gestión y puede acceder a los datos de todos los sensores y realizar un análisis más exhaustivo de lo que ocurre en la red. En caso de futuros ataques, considere instalar un único sensor que supervise y rastree todo el tráfico.

Luego hay un servidor de base de datos donde se almacenan los registros y alertas de los sensores (que también se almacenan directamente en el ordenador de gestión o en un sistema de archivos distribuido si hay varios registros). Entre las precauciones que suele tomar el IPS se encuentran descartar paquetes o una sesión con actividad objetable, reiniciar la sesión, bloquearla y añadirla a una lista negra.

Es un sistema de código abierto propiedad de una fundación iniciada por la comunidad (OSPF, Open Information Security Foundation).

Suricata suele complementar (y no sustituir) a otras herramientas, como los cortafuegos. Dado que proporciona capacidades IDPS (combinando funciones de detección y prevención en un solo mecanismo), puede cubrir la mayoría de las principales vulnerabilidades de red relacionadas con la transmisión de mensajes desde áreas internas de una organización a áreas externas y zonas desmilitarizadas (Locicero Giorgio, 2020).

Existen cuatro tipos de IDPS (de red, inalámbrico, de análisis de comportamiento de red, de host), y los escenarios de ataque se basan en las técnicas de red y host más utilizadas por otros IDPS como Suricata y Snort.

OSSEC. Los IDS, como el sistema de detección de intrusiones basado en host (OSSEC) de código abierto, son una herramienta importante para que el administrador del sistema prevenga posibles amenazas a los sistemas y los datos. Un OSSEC IDS consiste en un administrador que analiza los registros de eventos de un grupo de agentes y los compara con ciertos estándares definidos en reglas. Si los eventos de los agentes de OSSEC coinciden con los patrones definidos en las reglas, el administrador de OSSEC emite acciones predefinidas que se aplican a los agentes durante un determinado periodo de tiempo. Ejemplos de estas acciones son bloquear una dirección IP potencialmente maliciosa o añadir una dirección IP a una lista específica de dispositivos restringidos durante un periodo de tiempo predefinido. Los registros de los agentes se eliminan tras su procesamiento en OSSEC Manager y se almacenan en cada agente. Hay determinados escenarios en los que OSSEC tiene restricciones en las detecciones de falsos positivos y falsos negativos (Teixeira et al., 2019).

Zeek. Es una plataforma altamente personalizable para el análisis de redes y la detección de intrusiones. Zeek puede personalizarse añadiendo scripts. Existen muchos scripts para Zeek que pueden detectar diferentes tipos de tráfico malicioso. Sin embargo, como los nuevos ataques se desarrollan a un ritmo rápido, escribir scripts para todos ellos es una tarea que lleva mucho tiempo. Se necesita una solución más general que complemente las capacidades de detección de Zeek. Aquí entra en juego el aprendizaje automático (ML): se puede entrenar un modelo ML en datos de tráfico malicioso y seguro. Este modelo puede utilizarse para detectar flujos de tráfico malicioso en los registros de Zeek. En trabajos anteriores se han realizado experimentos similares, pero con diferentes IDS, diferentes conjuntos de datos y sin integración de NIDS en tiempo real [6] ni clasificación ML en tiempo real. La novedad de este trabajo radica en la combinación particular de Zeek, el conjunto de datos seleccionado y la detección ML en tiempo real (Gustavsson, 2019).

Zeek funciona de forma diferente a Snort y Suricata en el sentido de que el analizador Zeek consiste en un motor basado en eventos que se activa en función de diversos eventos operativos. Estos eventos se envían a Zeek: un dominio de escenarios donde se analizan las características del tráfico mediante el lenguaje de escenarios de Zeek, que genera registros y alertas de actividad sospechosa o interesante. El lenguaje de scripts es totalmente Turing, lo que convierte a Zeek en una plataforma de análisis de tráfico altamente personalizable.

2.6. Inteligencia Artificial

La informática tiene una subdivisión conocida como Inteligencia Artificial (IA). Se ocupa del desarrollo de programas informáticos para realizar tareas que, de otro modo, requerirían inteligencia humana. Los algoritmos de IA pueden centrarse en el aprendizaje, la percepción, la resolución de problemas, la comprensión del lenguaje y/o el razonamiento lógico (Khaled AlSedrah Miriam, 2017).

La necesidad de la IA crece día a día. Desde su aparición, la IA ha provocado rápidos cambios en la tecnología y los negocios. Los informáticos predicen que en 2020 el 85% de las interacciones con los clientes se producirán sin intervención humana. Esto significa que las peticiones humanas sencillas dependerán de los ordenadores y la IA, como el uso de Siri o Galaxy para averiguar la temperatura del aire (Mohammed Ziyad, 2019).

2.6.1. Capacidades de la Inteligencia Artificial

- La inteligencia artificial puede hacer predicciones y adaptarse mediante algoritmos que reconocen patrones en grandes cantidades de información.
- Al tomar decisiones por sí misma, la IA puede mejorar la inteligencia humana, proporcionar información y aumentar la productividad.
- La IA aprende constantemente y utiliza algoritmos para crear modelos analíticos. Basándose en estos algoritmos, la tecnología de IA aprende a realizar tareas a través de innumerables pruebas y errores.
- En el futuro, la IA será una herramienta que permitirá a los humanos replantearse la forma en que analizan los datos e integran la información, y luego utilizar esa información para tomar mejores decisiones.
- La IA puede moverse y percibir objetos.

El aprendizaje automático ha evolucionado hasta convertirse en una rama de la inteligencia artificial (IA) que permite a investigadores y desarrolladores crear sistemas capaces de aprender de los datos y mejorar su rendimiento a medida que se acumula más información. Hoy en día, el aprendizaje automático se utiliza en una amplia gama de campos y se ha convertido en una herramienta importante para la toma de decisiones, la detección de fraudes, el reconocimiento de

patrones, el diagnóstico médico y muchas otras áreas. Por ello, en este artículo se analizarán algunos ejemplos de trabajos realizados con aprendizaje automático para entender cómo esta tecnología puede mejorar diversas aplicaciones y cómo se han desarrollado soluciones innovadoras utilizando esta tecnología.

2.6.2. Clasificación de la Inteligencia Artificial

Machine Learning. Las técnicas de machine learning (ML) permiten a los sistemas aprender de su propia experiencia. El aprendizaje automático es la capacidad de un sistema para adquirir e integrar una serie de elementos mediante una observación exhaustiva y para mejorar y ampliar sus capacidades adquiriendo nuevos contenidos en lugar de programarlos.

El aprendizaje automático es una amplia disciplina con teorías estadísticas básicas sobre los procesos de aprendizaje, ha desarrollado algoritmos de aprendizaje utilizados en aplicaciones comerciales (reconocimiento del habla, visión por ordenador) y creado la minería de contenidos que descubre patrones ocultos en cantidades cada vez mayores de datos basados en la web.

Estas tecnologías organizan los recursos existentes y extraen nuevos conocimientos mediante la captura inteligente de metadatos y la extracción de conclusiones a partir de ellos. Los programas de aprendizaje han logrado resultados muy diversos, desde la memorización trivial hasta la creación de teorías científicas totalmente nuevas, y tienen potencial para mejorar constantemente y ser cada vez más eficientes y eficaces (Woolf, 2009).

Aprendizaje Profundo. El aprendizaje profundo es una de las áreas de la informática que más rápido está creciendo. Aprendizaje profundo es un algoritmo basado en redes neuronales artificiales optimizados para procesar datos no estructurados como imágenes, audio, vídeo y texto. Aunque las técnicas de aprendizaje profundo se desarrollaron ya a mediados de la década de 1980,

su verdadero potencial no se ha hecho realidad hasta los últimos cinco años (Kotu & Deshpande, 2019).

El aprendizaje profundo y el campo de la inteligencia artificial han sido palabras de moda durante varios años y su popularidad no ha dejado de aumentar. El aprendizaje profundo es un campo del aprendizaje automático inspirado en el funcionamiento de nuestro cerebro. Utilizando una red de capas llamadas redes neuronales, cada capa toma gradualmente información diferente y luego produce una salida. El aprendizaje profundo se está convirtiendo gradualmente en una fuerza real en el campo del aprendizaje automático, especialmente cuando se trata de utilizar grandes cantidades de datos (Janeczko & Srivastava, 2022).

Sistemas Expertos. Los sistemas expertos forman parte del campo más amplio de la inteligencia artificial (IA). Utilizan un enfoque simbólico para representar el conocimiento y modelar el proceso que utilizan los expertos para resolver problemas. Para entender qué es la IA y cómo funciona, primero tenemos que introducir algunos conceptos básicos sobre la pericia y qué hace que una IA sea experta. Dado que uno de los objetivos de la IA es la adquisición de conocimientos, hablaremos del proceso de adquisición de conocimientos. El conocimiento debe representarse principalmente en términos de reglas de producción, pero existen otras representaciones, y cada problema tiene una correspondencia natural con una o más representaciones del conocimiento. Para que el conocimiento sea útil, debe aplicarse, una función del mecanismo de inferencia, el cerebro. Los desarrolladores de aplicaciones deben comprender los factores que hacen que un sistema experto tenga éxito o no (Aronson, 2003).

2.6.3. Algoritmos de Detección

Los modelos de detección de anomalías se establecen de acuerdo con la naturaleza de la entrada, clase de anomalía, etiquetas de los datos, o salida del que presenta el modelo. Por otro lado, la detección de intrusiones es un desafío para la seguridad de redes debido a que su objetivo principal es identificar el tráfico inusual o anómalo o directamente de ataques a la seguridad de la red. Permitiendo así que se desarrollen los sistemas de detección de intrusiones donde se aspira a que se emitan alertas tempranas ante intrusiones previniendo o minimizando el daño (López Ávila et al., 2019).

A continuación, en la Tabla 1 se presentan algunos de los algoritmos que se utilizan para la detección de anomalías o intrusiones dentro de un sistema:

Tabla 1*Algoritmos de detección.*

Algoritmo	Descripción	Ejemplos de modelos	Ventajas
Basados en reglas	Este tipo de algoritmo busca patrones continuos dentro de un grupo de datos, los cuales pueden ser usados para la detección de asociaciones entre elementos (Heaton, 2016).	Apriori FP-Growth	Se pueden identificar el comportamiento malicioso de acuerdo con las reglas establecidas, lo que le hace que su detección sea predefinida.
Basados en anomalías	El algoritmo basado en anomalías se basa en que se identifiquen instancias anómalas consideradas como intrusiones en un grupo de datos, estos datos pueden ser de diferentes tipos ya sean tabulares, espaciales o complejos (Effrosynidis, 2020).	Isolation Forest One-Class SVM DBSCAN	Al permitir que el tipo de datos con el que se pueda trabajar sea complejo, puede realizar la detección de nuevas intrusiones, debido a que se enfoca en el comportamiento anómalo y no solo en patrones que ya son predefinidos.
Basados en redes neuronales	Las redes neuronales su base son modelos de aprendizaje profundo, los que sirven para la detección de anomalías en grupo de datos complejo, requiriendo de esta manera un entrenamiento minucioso (Yan, 2021).	Redes Neuronales Convolucionales (CNN) Redes Neuronales Recurrentes (RNN)	Este algoritmo puede capturar las características de un grupo complejo y adaptarse a cambios, por lo que permite que se pueda detectar patrones de comportamiento malicioso.

2.7. Trabajos Relacionados con Aplicaciones de Inteligencia Artificial

La combinación de IA y WSN puede mejorar significativamente la eficacia y precisión de las WSN en diversas aplicaciones. Muchos trabajos abordan dicha combinación y proponen diversos modelos y algoritmos de IA para mejorar el rendimiento de las WSN. En esta sección, analizamos varios trabajos que utilizan IA en las WSN y proporcionamos una visión general de los métodos, modelos y técnicas utilizados.

- **Díaz (2022) “Sistema de monitoreo inteligente basado en tecnología IoT e inteligencia artificial aplicado a la crianza de alevines, en la parroquia el Playón de San Francisco perteneciente al cantón Sucumbíos provincia Sucumbíos”**

Menciona la aplicación de un algoritmo de lógica difusa para controlar la calidad del agua en un sistema inteligente de vigilancia de la pesca, especialmente en lagos de trucha arco iris. La lógica difusa se utiliza en el sistema para informar al usuario cuando los parámetros de calidad del agua pasan a ser críticos. El algoritmo de lógica difusa utiliza 27 reglas difusas basadas en los conocimientos del personal del laboratorio para determinar cuándo los parámetros de calidad del agua se encuentran en estado crítico. La precisión del sistema es del 100% y el porcentaje de aciertos es del 94,4%, como muestra la tabla de errores (Díaz, 2022). El uso de la lógica difusa en este sistema alerta tempranamente al usuario de las condiciones críticas de la calidad del agua, lo que ayuda a prevenir la muerte de peces y mejorar la eficiencia de la piscicultura.

- **Salazar Cárdenas (2019) “Diseño de un Sistema de Riego Inteligente para Cultivos de Hortalizas Basado en Fuzzy Logic en la Granja La Pradera De La Universidad Técnica Del Norte”.**

En el proyecto se emplea la lógica difusa en el desarrollo de un sistema de riego inteligente para cultivos de hortalizas. La lógica difusa es un método de inteligencia artificial utilizado para tratar la incertidumbre y la imprecisión en la toma de decisiones. En este caso, se utiliza para controlar el momento de riego de cultivos de hortalizas en función de factores ambientales. El sistema difuso tiene entradas y salidas, donde las entradas son factores ambientales que afectan a la producción de hortalizas, como la temperatura, la humedad del aire y del suelo, la radiación solar y el viento. Las salidas corresponden al tiempo de riego necesario para mantener unas condiciones óptimas de cultivo.

El sistema de riego inteligente basado en la lógica difusa se comparó con el riego manual por goteo y se comprobó que el uso de la tecnología de riego basada en la lógica difusa reduce el consumo de agua. Esto se debe a que el sistema de riego inteligente permite un uso más eficiente del agua al ajustar el tiempo de riego en función de las condiciones ambientales de las plantas (Salazar Cárdenas, 2019).

- **Naula López (2021) “Diseño e Implementación de un Sistema de Detección de Intrusiones para redes WiFi usando herramientas de Big Data y Machine Learning”.**

El artículo describe un proyecto que utiliza Big Data y herramientas de aprendizaje automático para implantar un sistema de detección de intrusiones en redes Wi-Fi. El sistema utiliza un modelo de aprendizaje automático de tipo random forest para clasificar el tráfico de red y distinguir entre tramas normales y tramas maliciosas generadas por intrusos. El sistema también utiliza herramientas de procesamiento de Big Data como Apache Spark, Kafka y Elasticsearch para analizar y visualizar los resultados en un cuadro de mando creado en el entorno Kibana. El modelo

de aprendizaje automático Random Forest se utiliza por su alta precisión en la detección de anomalías en el tráfico Wi-Fi.

El modelo Random Forest es un conjunto de árboles de decisión que se combinan para mejorar el rendimiento. En Spark, este modelo admite la clasificación binaria, multiclase o de regresión y puede manejar características categóricas y numéricas sin necesidad de normalizar o escalar las características. Al entrenar árboles de decisión, cada árbol se entrena de forma independiente y aleatoria para que cada árbol sea diferente de los demás. En la clasificación, la predicción final se basa en la decisión mayoritaria (Naula López, 2021).

Según cada artículo mencionado, no se puede determinar qué método de aprendizaje automático o inteligencia artificial es el más adecuado en distintos contextos, ya que depende de las características y necesidades específicas del proyecto. Cada uno de estos métodos tiene ventajas e inconvenientes y debe seleccionarse en función de las características del problema a resolver y de los datos disponibles.

CAPÍTULO III: DISEÑO DEL IDS

El diseño de IDS en redes de sensores inalámbricas (WSN) puede suponer un reto, especialmente cuando se utilizan técnicas de inteligencia artificial para mejorar la eficiencia y precisión de la detección de intrusos. El propósito de este capítulo es presentar un enfoque para desarrollarlo basado en la metodología Agile.

El capítulo describe los principales aspectos del diseño de IDS para WSNs, incluyendo las fases de desarrollo que se aplican de la metodología establecida dentro del proyecto. En la fase 1 se establece que, el método se determinará sobre la base de una evaluación de los procedimientos a realizar, centrándose en la revisión de los conceptos de ataque a las WSN pertinentes y potencialmente peligrosos, es decir, se espera que el análisis proporcione información suficiente para su integración en el sistema, etapa implementada en secciones anteriores.

La fase 2, basándose en el análisis de los ataques y el diseño de la red, se desarrolla un proceso para mejorar y optimizar la estructura resultante utilizando conceptos y técnicas de aprendizaje automático, además de establecer los requerimientos principales para el lanzamiento de la etapa. Una vez implementada la configuración, puede probarse a fondo frente a las capacidades existentes y, por último, mejorarse en términos de funcionalidad y capacidad de despliegue del IDS en la red.

3.1. Requerimientos

La sección presentara los parámetros necesarios para el cumplimiento de los objetivos e implementar de manera eficiente la comunicación de la red, permitiendo el cumplimiento del alcance establecido.

Ahora se muestra un análisis de requerimientos basado en el estudio de documentación bibliográfica, información relacionada con el uso de softwares de simulación, ataques y modelos de inteligencia artificial aplicables al proyecto. Ejecutando este proceso tanto con el uso de la norma (ISO/IEC/ IEEE 29148, 2018) como de la Arquitectura de Requisitos del Sistema (SyRA) considerando las adaptaciones necesarias para el proyecto a desarrollar, teniendo en cuenta el uso del **Anexo A** realizado con la finalidad de la determinación de parámetros que se establecen dentro de la norma.

A continuación, se menciona los requerimientos definidos por el estándar ISO/IEC/IEEE 29148:2018:

- **StRS**: Especificación de requerimientos de las partes interesadas (Stakeholders).
- **SyRS**: Especificación de requerimientos del sistema.
- **SRS**: Especificación de requerimientos de software.
- **SyRA**: Especificación de requerimientos de arquitectura.

3.1.1. Especificación de los Requisitos de las Partes Interesadas (StRS)

Determina los requisitos de las partes interesadas, incluidos usuarios, clientes, patrocinadores y otros grupos interesados. Se centra en las necesidades de las partes interesadas y los requisitos derivados de ellas, y garantiza que los requisitos del sistema sean coherentes con los requisitos empresariales y los objetivos estratégicos.

La definición establecida en secciones anteriores permite especificar a los beneficiarios que usarán la herramienta desarrollada como se muestra en la Tabla 2.

Tabla 2

Definición de los Stakeholders.

Nro.	Stakeholders	Detalles
StRS1	Ing. Fabián Cuzme, MSc.	Tutor de tesis
StRS 2	Ing. Luis Suárez, MSc.	Asesor de tesis
StRS 3	Velastegui Jhoselyn	Estudiante

Nota. La tabla establece los stakeholders que pueden ser admitidos dentro de la implementación del sistema.

3.1.1. Especificación de los requisitos del sistema (SyRS)

Esta especificación describe los requisitos del sistema, incluidos los requisitos funcionales y no funcionales, las restricciones del sistema y los criterios de aceptación. La SyRS describe el comportamiento del sistema, las interfaces con otros sistemas y las restricciones que deben tenerse en cuenta en el diseño. La SyRS constituye la base para el desarrollo de una solución de sistema y es utilizada por los desarrolladores para diseñar, construir y probar el sistema.

Considerando esos aspectos principales se observa la Tabla 3 de requerimientos del sistema:

Tabla 3

Definición de los requerimientos del sistema.

Nro.	Requerimientos	Prioridad		
		Alta	Media	Baja
SyRS1	El sistema de simulación debe virtualizar la red WSN, además de que cada nodo y que permita la comunicación.	X		
SyRS2	El sistema de simulación debe permitir virtualizar la comunicación inalámbrica entre nodos sensores.	X		
SyRS3	El sistema de simulación debe monitorear el comportamiento de la red.	X		
SyRS4	El sistema de simulación deber recibir la alerta en caso de detectar un intruso.	X		
SyRS5	El sistema de simulación debe incluir interoperabilidad con otro tipo de softwares para realizar la implementación de un IDS.	X		
SyRS6	El sistema de simulación debe aplicar inteligencia artificial comprobando que aumente la precisión de la detección de intrusos.		X	
SyRS7	El sistema de simulación debe tener un tiempo de repuesta considerablemente bajo hablando de detección de intrusos y la emisión de alertas.		X	
SyRS8	El sistema de simulación debe permitir que se añada configuraciones para el ataque que se requiere implementar.	X		
SyRS9	El sistema de simulación deber ser capaz de implementar protocolos de comunicación.		X	
SyRS10	El sistema de simulación tiene que ser compatible con el hardware de la persona que lo implemente.		X	
SyRS11	El sistema de simulación debe permitir realizar pruebas de funcionamiento.	X		

Nota. En la tabla se muestran los requerimientos necesarios para establecer el mejor software para virtualizar la WSN desarrollado para el proyecto.

3.1.2. Especificación de Requisitos Del Software (SRS)

Describe los requisitos del software y se utiliza en la aplicación de la norma ISO/IEC/IEEE 12207. La SRS define los requisitos de software que deben cumplirse para satisfacer el sistema. El SRS sirve de base para el diseño y desarrollo de software y también lo usan los desarrolladores al implementar y probar el software.

3.1.3. Arquitectura de Requisitos del Sistema (SyRA)

Proporciona la estructura y organización de los requisitos del sistema. SyRA ayuda a definir la estructura del sistema y especifica las relaciones entre los distintos componentes del sistema. SyRA se centra en la arquitectura de requisitos del sistema y se utiliza para definir la estructura del sistema, especificar las interfaces entre los componentes y definir los requisitos de comportamiento y rendimiento. SyRA se utiliza como herramienta de planificación, gestión y control del proceso de desarrollo de requisitos. También puede utilizarse para verificar y validar los requisitos del sistema e identificar posibles problemas o riesgos durante el desarrollo del sistema (Vallero et al., 2019).

Para ello, se identifican los factores específicos que deben tenerse en cuenta en el desarrollo de un sistema, incluidos los conceptos generales que influyen en el entorno en el que se utilizará el sistema.

- Conjunto de dispositivos de red virtualizados, como enrutadores y conmutadores, que se ejecutan como software en una máquina de simulación. Estos dispositivos pueden ser necesarios para simular la estructura de red deseada y las conexiones entre los nodos de la red de sensores.

- Una plataforma de virtualización, como VirtualBox o VMware, que permite ejecutar diversos sistemas operativos en un único equipo de simulación. Esto puede ser útil para simular nodos sensores individuales con diferentes configuraciones y sistemas operativos.
- Una serie de herramientas informáticas, como Cooja, Mininet, MATLAB o NS-2/NS-3, que permiten modelar redes de sensores inalámbricos y recopilar datos para la simulación.

Una vez definidos los parámetros principales para el funcionamiento del sistema, permitiendo aplicar así el establecimiento de dichos requerimientos en la Tabla 4.

Tabla 4

Requerimientos de arquitectura del sistema

Nro.	Requerimientos	Prioridad		
		Alta	Media	Baja
SyRA1	El sistema debe tener nodos inalámbricos y un gateway para permitir la conexión entre ellos.	X		
SyRA2	La detección de intrusos debe tener un tiempo de respuesta de manera inmediata.	X		
SyRA3	Las WSNs deben recopilar información constantemente.			X
SyRA4	La aplicación de la inteligencia artificial permitirá que la detección de intrusos sea más rápida y aprenda por el ataque recibido para futuras detecciones.	X		
SyRA5	Las pruebas sobre el sistema deben reflejar que se envió alertas de intrusión a la red.	X		
SyRA6	La WSN debe soportar el aumento de nodos y el trabajo en diferentes entornos.		X	
SyRA7	Se garantiza que la red es eficiente en aspectos de uso de recursos aumentando el rendimiento y minimizando el consumo de energía.			X

Nota. En la tabla se establecen varios de los parámetros que se necesitan para la elección del software.

La combinación de todos los elementos garantiza que la arquitectura esté diseñada para integrarse con el software de simulación, de modo que todos los procesos puedan funcionar con eficiencia y eficacia.

3.2. Selección de Software de Simulación

La simulación de redes inalámbricas de sensores (WSN) es una herramienta importante para desarrollar y evaluar soluciones a diversos problemas. El uso de la inteligencia artificial para detectar intrusiones en las WSN es un área de investigación que ha recibido mucha atención en los últimos años.

En este contexto, la selección de un software de simulación adecuado se convierte en una cuestión fundamental para la correcta implementación de soluciones de detección de intrusiones en WSNs simuladas utilizando IA.

En este trabajo se analizan herramientas de simulación según sus características para seleccionar el software que mejor se adapte a los requisitos de la arquitectura en desarrollo teniendo en cuenta el **Anexo A** donde se encuentra información específica para esta acción.

En la Tabla 5 se ve la validación de cada requerimiento establecido para el sistema a desarrollar, considerando que la base de los aspectos a tratar son los correspondientes a la Tabla 3.

Tabla 5

Verificación de los criterios para el sistema a implementar conjunto con los softwares de simulación que se analizaron.

Requerimientos del sistema (SyRS)	OMNeT++	Mininet	NS-2	NS-3	NetSim	Cooja
SyRS1	X	X	X	X	X	X
SyRS3	X	X	X	X	X	X
SyRS4	X	X	X	X	X	X
SyRS4	-	X	-	-	-	-
SyRS6	-	X	X	X	X	X
SyRS7	X	X	X	X	X	X
SyRS8	X	X	X	X	X	X
Total	5	7	6	6	6	6

Nota. La tabla se encuentra realizada en base al **Anexo A** del documento, donde se observar que cada “X” es una representación de que si cumple con ese parámetro. De acuerdo con los resultados obtenidos se establece que las plataformas con la mejor clasificación son Mininet, NS-2/NS-3, NetSim y Cooja según los parámetros considerados importantes que deben ofrecen cada uno de ellos, por otro lado se encuentran la plataforma como OMNeT++ considerado con una valoración menor a comparación de los primeros mencionado estos se diferencian en cuestiones como el tiempo de respuesta ya que se consideran programas que necesitan de recursos para la virtualización de las redes como para su instalación, donde se debería tener en cuenta el uso de los recursos disponibles.

Una vez estableciendo los parámetro principales se establece como programa o software principal para la simulación es Mininet, dado que se determina que es un programa completo donde se puede usar varios recursos para la implementación de la simulación con los ajustes que se

necesitan además de que, también se logra incluir herramientas externas para completar el sistema que se desea desarrollar entre ellos se puede mencionar el uso de la inteligencia artificial y las configuraciones para la detección de intrusos.

3.3. Selección de Hardware para Simulación

En esta sección se analiza un aspecto importante para la instalación del programa o software simulador que se emplea para la virtualización de una WSN, por lo que se determinan requerimientos mínimos necesarios para que se garantice el funcionamiento de manera correcta del sistema.

Mininet es un programa que se ejecuta sobre sistemas Linux, por lo que para su instalación es necesario que el ordenador tenga este tipo de sistema operativo o bien una máquina virtual que permita su emulación (Ojeda Guerra Carmen Nieves, 2020). En este caso el ordenador si cuenta con el S.O. Linux de manera nativa.

Considerando las posibilidades del uso de los softwares establecidos se determinan los requerimientos de manera general con el objetivo de que estas sirvan para el elegido, además de que este aspecto se realiza basado de acuerdo con la base bibliográfica que se recopiló anteriormente para la instalación del software, siendo así que ahora se presentan las características principales dentro de la Tabla 6.

Tabla 6

Especificaciones de los requerimientos mínimos para el software.

Especificaciones del ordenador	
Procesador	Intel (R) Core (TM) i5-6300U
Memoria RAM	16 GB
Disco	SAMSUNG MZNTY256 (256GB)
CPU	2 núcleos
Especificaciones para el sistema	
Memoria RAM	8 GB – recomendable 16GB
CPU	2 núcleo
Almacenamiento para instalación	35 GB
Almacenamiento para complementos	Desde 50 GB o más

Nota. Las características mencionadas son las recomendables para que el software opere de manera correcta, pero si el usuario opta por parámetros superiores a las establecidas, funciona sin problemas.

3.4. Selección de Ataque

En este apartado se establece la relación con el ataque que se utiliza en el diseño del sistema, considerando que en el Capítulo II se dan a conocer bases sobre algunos de ellos, que existen y pueden implementarse, por lo que, ahora se analiza el ataque realizado en una WSN.

3.4.1. Identificación de Ataque

La identificación del ataque se muestra en la Tabla 6 donde se encuentra una breve explicación de lo que puede realizar cada uno de ellos, además de las contramedidas o mitigaciones que se pueden implementar.

Por otro lado, para la construcción de la tabla se considera el **Anexo B**, donde se explica con mayor detalle respecto a su función y las capas en las que se pueden implementar.

Tabla 7

Características principales para la identificación del ataque.

Tipo de Ataque	Activo/Pasivo	Descripción	Contramedidas
Denegación de servicio distribuido (DDoS)	Activo	Crea congestión en la comunicación de la red. Consumo de energía, reduciendo a vida útil del nodo.	Detección de intrusos monitoreando el tráfico de la red y seguridad.
Denegación de servicio	Activo	Inundar las redes con una cantidad masiva de tráfico malicioso o solitudes. Agotar recursos para interferir en el funcionamiento correcto de la red.	Detección y mitigación de ataques identificando patrones o comportamientos maliciosos. Firewall permitiendo el cloqueo de tráfico malicioso antes de llegar a la red.
Inundación	Activo	Agota los recursos de la red tanto para ancho de banda como de energía. Lanzamiento de diversos nodos y aumenta su eficacia.	Sistema de detección de intrusos monitoreando el tráfico de la red e identificar los patrones para un comportamiento malicioso.
Falsificación de identidad	Activo	Técnicas de suplantación de identidad en n nodo de la red Envío de información maliciosa a los nodos de la red.	Sistema de autenticación y autorización, validando la identidad de dispositivos y usuarios para el acceso a la red. Técnicas de cifrado.
Retransmisión	Activo	Uso en combinación con otros ataques aumentando su eficacia. Intercepta y reenvía paquetes de datos en los nodos.	Técnicas de autenticación y cifrado. Sistemas de control de acceso.

Nota. En la tabla se observan las características principales de los ataques más usados en una red WSN.

Dadas las diversas modalidades de ataque habituales en las redes inalámbricas de sensores (WSN), en este proyecto se ha optado por un ataque activo de denegación de servicio (DoS) debido a sus ventajas. Este tipo de ataque permite evaluar aspectos importantes de la red, como su disponibilidad. Una WSN tiene un nodo central que procesa la información. Al exponerlo a un ataque DoS, se puede comprender el impacto en el servicio. Al elegir este tipo de ataque, podemos hacernos una idea clara del rendimiento de la red y de su capacidad de recuperación ante eventos perjudiciales.

3.5. Selección de Algoritmo de Detección

La selección del algoritmo de detección que será aplicado por medio de la inteligencia artificial se establece con respecto a los requisitos preestablecidos para el funcionamiento del sistema, basado en la norma ISO/IEC/IEEE 29148:2018. Un elemento principal es la elección del algoritmo por lo que, la Tabla 8 se fundamenta en base al **Anexo C** dando a conocer algunos de los tipos de algoritmos para la aplicación dentro del sistema:

Tabla 8

Algoritmos de detección para la identificación del modelo a implementar.

Características	Apriori	FP- Growth	Isolation Forest	One- Class SVM	DBSCAN	CNN	RNN
Simplicidad de implementación.	-	-	1	-	-	-	-
Procesamiento para la carga de trabajo al ejecutar el algoritmo.	-	1	1	-	-	-	-
Precisión para la identificación de patrones anómalos.	-	-	1	1	1	-	-
Capacidad de la carga de datos para su análisis.	-	1	1	1	1	1	1
Interoperabilidad con otros sistemas o tecnologías dentro del entorno de implementación.	1	1	1	-	1	-	-
Capacidad de trabajar con diversos tipos de datos y formatos.	-	1	1	-	-	1	1
Total	1	4	6	2	3	2	2

Nota. En la tabla se observan las características principales que deben cumplirse para la implementación dentro del sistema, de acuerdo con lo que se muestra en la tabla se considera que, si contiene la característica siempre y cuando se encuentre valorado con el número 1, mientras que si se encuentra marcado con un guion se establece que no cuenta con esa característica.

Según los algoritmos que se pueden aplicar para la detección se establece que la opción más viable es Isolation Forest debido a que este se puede trabajar con los diversos tipos de datos, también presenta una gran eficiencia para la detección de anomalías debido a que construye árboles de decisión aleatorios para así separar las instancias anómalas.

Dentro de la Tabla 8 presentan la puntuación más alta dentro de la comparación con otro tipo de modelos, que pueden mostrar en su mayoría la interoperabilidad con otras herramientas, pero ese aspecto no es suficiente para que sirva dentro del sistema desarrollado.

3.6. Selección de Sistema Operativo

El sistema operativo es un punto crucial dentro del desarrollo del sistema, debido a que este debe ajustarse a los requisitos que fueron establecidos previamente. La Tabla 9 aborda los tipos de S.O. que podrían usarse para el desarrollo del sistema, en aspectos como la estabilidad, personalización, disponibilidad, seguridad y el costo del uso de los mismos, siendo el S.O. Linux el que se destaca debido a que este permite que se trabaje acorde con la compatibilidad y requisitos de las herramientas antes mencionadas.

Tabla 9

Descripción de los sistemas operativos con respecto a los requerimientos establecidos para la implementación del sistema.

Característica	Linux	macOS	Windows
Compatibilidad con Mininet	Linux es el sistema operativo preferido para Mininet, ya que Mininet se desarrolló originalmente para Linux y ofrece compatibilidad total con Linux.	Mininet también puede ejecutarse en macOS, pero requiere ajustes adicionales, como la instalación de una máquina virtual Linux (por ejemplo, a través de VirtualBox) o el uso de un contenedor.	Aun cuando Mininet puede ejecutarse en Windows utilizando una máquina virtual o emulación, no se recomienda como plataforma nativa y puede requerir una configuración más compleja.
Estabilidad y Rendimiento	Linux es conocido por su estabilidad y rendimiento en servidores y sistemas de red. Con Mininet se puede trabajar de forma eficiente, especialmente en entornos de servidor.	macOS es conocido por su estabilidad y rendimiento para sistemas de escritorio y desarrollo, pero puede no ser tan eficiente como Linux para ejecutar simulaciones de red.	Windows es adecuado para aplicaciones de escritorio y desarrollo, pero puede tener limitaciones de rendimiento y estabilidad cuando se ejecutan simuladores de red como Mininet.
Facilidad de Personalización	Linux tiene una amplia gama de distribuciones y puede adaptarse fácilmente a los	El sistema macOS tiene limitaciones de personalización debido	Windows es menos flexible en términos de personalización,

	requisitos específicos al entorno controlado especialmente en de modelado de las por Apple. comparación con Linux. WSN.
Disponibilidad de Software	Linux cuenta con una amplia gama de software y herramientas de código abierto ideales para desarrolladores y simuladores, incluidas herramientas de gestión de redes. macOS cuenta con un sólido ecosistema de software, pero algunas herramientas específicas de red pueden no estar disponibles o requerir una configuración adicional. Windows ofrece una amplia gama de software, pero algunas aplicaciones de código abierto y herramientas de red pueden no estar tan disponibles como en Linux.
Seguridad y Mantenimiento	Linux cuenta con una activa comunidad de desarrolladores que publica periódicamente actualizaciones y parches de seguridad. macOS es conocido por su seguridad, pero la gestión de actualizaciones está sujeta a las políticas de Apple. Windows ofrece actualizaciones de seguridad periódicas, pero es más atractivo para los atacantes debido a su elevado número de usuarios y a su popularidad.
Costo	Linux es gratuito y de código abierto, lo que lo convierte en una opción rentable. macOS es propiedad de Apple y a menudo requiere hardware de Apple, que puede ser caro. Windows requiere el pago de licencias y el coste del hardware compatible puede variar.

Nota. La tabla presenta varias características que sirven para la elección del S.O. con respecto al uso de las herramientas adicionales para el desarrollo del sistema.

Como se mencionó anteriormente, la elección del S.O. recae en Linux, debido a su capacidad para ofrecer estabilidad, seguridad y flexibilidad, características fundamentales aplicables a una diversidad de entornos, desde estaciones de trabajo hasta servidores. Entre las diversas distribuciones de Linux, destaca Ubuntu por su excepcional facilidad de uso, apoyo de comunidad activa de usuarios, integración con diversos tipos de hardware, versiones de soporte. Además de su extenso repositorio de software y la integración con el ecosistema Debian y la constante innovación. En comparación de otras distribuciones Ubuntu brinda una experiencia de usuario robusta y confiable, convirtiéndole en una elección ideal para la implementación de sistemas.

3.7. Conectividad Inalámbrica Dentro del Entorno de Simulación

En el contexto de las WSNs, donde la conectividad inalámbrica es importante, el estándar IEEE 802.15.4 surge como fundamento crucial. Debido a que este se establece como requerimiento para la virtualización de la red. Por lo que, en esta sección se ofrecen los conceptos fundamentales del estándar, destacando su relevancia dentro del sistema desarrollado.

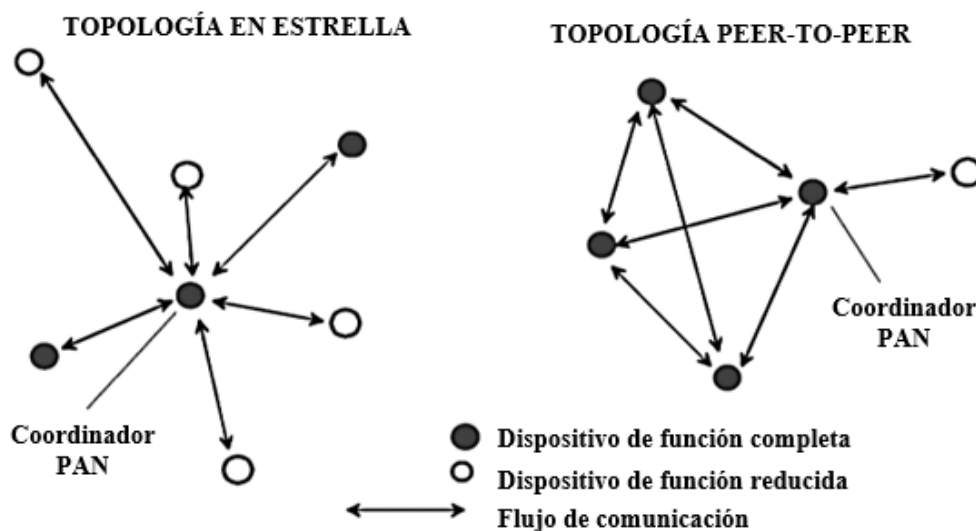
El estándar IEEE 802.15.4 menciona que existen dos tipos de dispositivos diferentes, uno que realice la función completa (FFD) y otro con la función reducida (RFD).

Los dispositivos FFD permite que actúe como coordinador de una red de área personal (PAN) o ya sea como coordinador, por otro lado, el dispositivo RFD no puede actuar como coordinador PAN, ni como coordinador, dado que se aplica en procesos simples, al referirse a que no se necesitan muchos datos, además de que solo se asocia como un único FFD a la vez (IEEE 802.15.4, 2020).

En este aspecto se debe tener en cuenta el tipo de aplicación, en la Figura 8 se aprecia las topologías en el caso de que sea una red IEEE 802.15.4 LR-WPAN la cual puede trabajar en dos tipos: estrella o peer-to-peer.

Figura 8

Ejemplos de topología en estrella y peer-to-peer.



Nota. Adaptado de IEEE 802.15.4 (2020).

La topología en estrella permite que su comunicación se realice entre dispositivos y que solo muestre un controlador central, siendo este llamado coordinador PAN. Cada dispositivo usado en cualquier tipo de topología tiene direcciones únicas o denominadas direcciones extendidas.

La topología peer-to-peer también presenta un coordinador PAN, pero el aspecto diferente a la topología estrella es que cualquier dispositivo podrá comunicarse con cualquier otro, considerando que se encuentre este dentro del radio de alcance, este tipo también permite la formación de redes complejas, como, por ejemplo, redes en malla.

3.7.1. Distribución de Nodos

En el estándar IEEE 802.15.4, los dispositivos se caracterizan por su corto alcance, baja tasa de bits y bajo consumo, además de que se encuentran limitados en potencia, memoria y disponibilidad de energía.

De acuerdo con lo que se establece anteriormente se puede determinar que algunos de los dispositivos generales o básicos son los nodos sensores que tiene como objetivo la recopilación de datos del entorno en donde se aplica, como por ejemplo temperatura, humedad, presión, entre otros, por otro lado, también está, el o los nodos de enrutamiento con los cuales se puede interactuar con la red dado que son los intermediarios entre los nodos de la sensores de la red, además de ayudar a que los datos puedan llegar a su destino y finalmente el uso de un Gateway, permitiendo la conexión con la red WSN y diversas redes, como, por ejemplo la Internet, es decir que ayudaría a que este pueda tener conexión con otros sistemas que trabajan de manera externa.

La tecnología LoRa y el estándar IEEE 802.15.4 son tecnologías que se pueden implementar dentro de las redes WSN, considerando las características y aplicaciones que se pueden llegar a conformar.

LoRa es una tecnología de comunicación inalámbrica en el que se sus características principales son su largo alcance y el bajo consumo de energía lo que en esencia necesita una WSN, considerando también que se aplica en ambientes de redes de sensores de baja potencia como por ejemplo la monitorización ambiental, agricultura, entre otros.

Algunos de los parámetros que se pueden configurar dentro de la tecnología LoRa son:

- **Modelo de propagación**

El modelo de propagación se refiere al modo en el que la señal LoRa se atenúa y propaga en el entorno. Algunos modelos de propagación habituales en LoRa son el modelo de Distancia Logarítmica, SIU, Okumura-Hata y Cost-231.

Según lo que mencionan algunos autores se concluye la Tabla 10 en la cual se explica el entorno, la distancia, aplicaciones y limitaciones con las que trabaja cada modelo de propagación que se menciona anteriormente (Khaled et al., 2022).

Tabla 10

Modelos de propagación que se pueden utilizar para redes inalámbricas.

Modelo de Propagación	Descripción	Distancia	Aplicaciones	Limitaciones
Distancia logarítmica	Este tipo de modelo se utiliza una ecuación logarítmica donde se puede calcular aproximadamente la pérdida de propagación en función de la distancia y otros parámetros.	Las distancias que se pueden manejar son variables, dado que serán establecidos de acuerdo con las condiciones específicas del ambiente en el que se requiere implementar	Redes de sensores. Internet de las cosas. El modelo Log-Distance se puede usar en entornos exteriores con LOS (Benavides Arrieta, 2019; Torres et al., 2016).	Los factores sobre el ambiente y terreno no se consideran.
Okumura-Hata	En el modelo Okumura-Hata se toma en cuenta factores como la distancia, frecuencia y la altura de la antena (Mollel & Kisangiri, 2014).		Redes de sensores urbanas. Aplicable para áreas urbanas y suburbanas.	Para el entorno en el que se trabaja no se recomienda áreas rurales.
Cost-231	Este modelo también trabaja con factores como distancia, frecuencia, altura de antena y el entorno en general (Trevino Cortes, 2003).		Redes de sensores. Aplicaciones de largo alcance.	En ciertos casos se puede necesitar de cálculos más complejos.

SUI	Se maneja para ambientes urbanos y suburbanos, donde se usa la pérdida de propagación causada por la presencia de obstáculos dentro del entorno (Galo Andy et al., 2023).	Redes de sensores.	Para el entorno en el que se trabaja no se recomienda áreas rurales.
-----	---	--------------------	--

Nota. En la tabla se muestra la información general de los modelos de propagación, dado que se debe considerar las características que se deben aplicar de acuerdo con el entorno en el que se requiere trabajar.

De acuerdo con lo que nos enseña la guía del programa de simulación que se está usando, es decir Mininet-WiFi soporta algunos modelos de propagación: Friis Propagation Loss Model, Log-Distance Propagation Loss Model, Log-Normal Shadowing Propagation Loss Model, International Telecommunication Union (ITU) Propagation Loss Model y Two-Ray Ground Propagation Loss Model, pero para usar otro tipo de modelo de propagación se deben efectuar los cambios necesarios para que este funcione dentro de la simulación.

Analizando los modelos mencionados dentro de la Tabla 9 se encuentran varios que pueden ser aplicados para redes inalámbricas, uno de ellos es el modelo de distancia logarítmica, volviéndose adecuado para predecir la intensidad de la señal de entornos exteriores donde puede que existan obstáculos afectando a la propagación de las ondas, también se considera como un modelo simple y comúnmente empleado, asimismo se puede implementar fácilmente en software de simulación como Mininet.

Considerando lo que se mencionó anteriormente en la guía de programa Mininet-WiFi existe la opción del modelo log-distance, además de que se establece un estudio del uso de Mininet-WiFi y

redes inalámbricas, donde se usa este modelo de propagación en conjunto con un modelo de movilidad para establecer situaciones prácticas (Panzuela Perez, 2022).

- **Distancia**

La distancia entre el transmisor y receptor es un aspecto que se considera para la propagación de la señal. Según la distancia que se aplica, su atenuación puede ser mayor, este factor se toma en cuenta dentro del cálculo del modelo de propagación para definir la atenuación que se espera de la señal a una determinada distancia.

LoRa presenta una alta sensibilidad de -137 dBm incrementando la disponibilidad de la red, además de que las señales penetran en muros de edificios e interiores profundos.

La distancia para los dispositivos LoRa entre el transmisor y receptor es aproximado a 3 km en la ciudad y de 6 km hasta 13 km considerando el entorno y las zonas urbanizadas. También se debe tener en cuenta que en la distancia de los dispositivos influyen factores como la propagación, el ancho de banda, potencia de transmisión, entre otros (Kuan, 2020).

Uno de los dispositivos Lora es el RN2903 de Microchip, a continuación, se detallarán las características generales de es módulo (Microchip, 2021):

- Banda de frecuencias: 902.000 MHz a 928.000 MHz.
- Método de modulación: FSK, GFSK y tecnología LoRa.
- Velocidad máxima de transmisión de datos por aire: 300 kbps con modulación FSK; 12500 bps con modulación de tecnología LoRa.
- Interfaz: UART.
- Alcance: Cobertura de hasta 15 km en zonas suburbanas y de hasta 5 km en zonas urbanas.

- Sensibilidad al 1% PER: -146 dBm.
- Potencia RF TX: Ajustable hasta máx. +18,5 dBm en la banda de 915 MHz.
- Generación de armónicos/Nivel de armónicos: Inferior a -70 dBm.
- Temperatura (funcionamiento): -40°C a +85°C.
- Temperatura (almacenamiento): -40°C a +115°C.
- Humedad: 10% ~ 90% sin condensación.

Según Burbano (2017) en su trabajo de “Implementación de una red de sensores inalámbricos LPWAN mediante módulos LoRa para el monitoreo de la calidad del agua en 2 ríos” efectúa pruebas de distancia con los módulos LoRa, permitiendo que de esta manera exista una distancia estimada o aproximada en el que se deberían colocar los nodos en exteriores para la simulación a implementar. En la Tabla 10 se presenta las distancias que los módulos LoRa pueden alcanzar según la línea de visión.

Tabla 11

Distancias que alcanzan los módulos Lora RN2903.

Metros	Sin línea de vista	Con línea de vista
1		
10		
20	Transmisión perfecta	
50		Transmisión perfecta
100	Transmisión con algunos fallos de comunicación	
200		
300		
400	Sin transmisión	
500		Transmisión con algunos fallos de comunicación

Nota. La tabla representa los datos obtenidos según las pruebas que realizo al desplazar los nodos en diversas distancias. Fuente: (Burbano, 2017).

Con la información de las pruebas que se realiza en el trabajo de Burbano (2017) se puede establecer que los nodos sensores pueden estar en un rango de distancia perfecto entre 1 y 50 metros, tanto para el caso de sin línea de vista como para de con línea de vista, rango ideal para que su comunicación sea de manera correcta, pero también se puede establecer un límite de distancia que en este caso es de 100 metros en el que se considera, por una parte con línea de vista de transmisión perfecta mientras que, para el caso de sin línea de vista este comienza a tener fallos de comunicación.

Determinando así que los nodos sensores con respecto al nodo coordinador puede oscilar entre el rango de 1 y 100 metros para su implementación dentro del simulador.

- **Ganancia de antena**

Este aspecto se considera como una medida con respecto a la eficacia con la que se comunica una antena en cierta dirección, esta se expresa en decibelios (dB) que puede ser negativa o positiva, cuando su configuración es positiva se dirige a una sola dirección para poder tener un mayor alcance. La ganancia en otras palabras se usa para la determinación de la intensidad de la señal a una cierta distancia.

En la Tabla 12 se encuentra las diferentes ganancias definidas de acuerdo con el entorno en que se implementa para interiores o exteriores (Yordanov, 2021) (Bertoldo et al., 2019).

Tabla 12

Ganancia de la antena definida según el entorno: interiores y exteriores.

Interiores	Exteriores
Antenas Omnidireccionales	Antenas Direccionales
Ganancia de la antena entre 0dBi a 5 dBi.	Ganancia de la antena entre 5 dBi a 12dBi.
Cobertura de señal amplia en varias direcciones.	Cobertura de la señal a una zona en específico.
Usado para entornos en espacios interiores que se encuentran cerca unos de otros.	Mayor distancia de transmisión, además de la penetración de la señal en ambientes abiertos.

Nota. La tabla hace referencia a los tipos de entornos que existe, es decir para interiores o exteriores, en los que se debe tener en cuenta que se definen de acuerdo con las variables de estos, como, por ejemplo, los obstáculos y de que sus valores son estimados para el tipo de antena que se puede implementar.

Dentro de este aspecto se debe considerar el tipo de antena que se puede implementar, como por ejemplo en las antenas yagi se basa en la propagación del campo electromagnético por medio de los directores y si se una cantidad considerable dirigida a una zona mayor será su ganancia, la antena de plato parabólico son de gran tamaño y se encuentran expuestas a factores del ambiente como el viento, entre otros (Felipe & Angel, 2020).

3.7.2. Diseño de la red

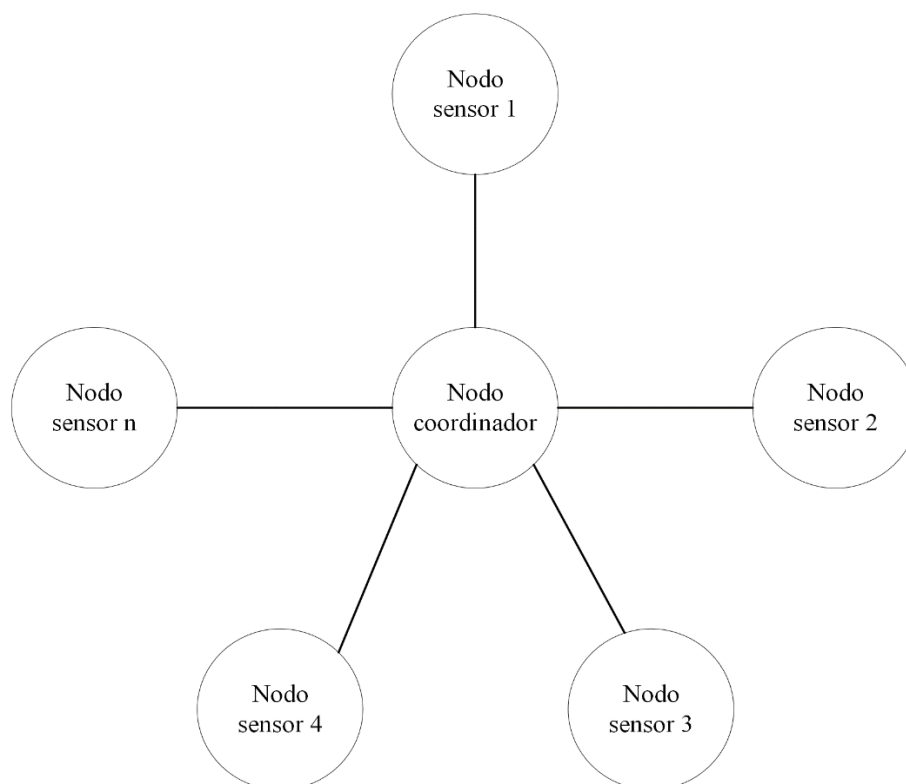
El diseño de la red consta de nodos con la función de nodo sensor y nodo coordinador, en este caso, la topología desarrollada será de tipo estrella considerada como la más simple, debido a que

utiliza un solo nodo coordinador PAN, por lo que cada nodo sensor se conecta directamente al nodo concentrador, esta topología también se encuentra limitada con respecto al área de cobertura con el alcance que tiene el nodo coordinador, pero como se menciona anteriormente se considera fácil de configurar e instalar.

La Figura 9 presenta un diagrama representando la topología de manera general, donde los n nodos sensores serán los que recogen la información del ambiente al que se aplique y el nodo coordinador que figura el centro de la red, además de recibir la información de los nodos sensores.

Figura 9

Distribución de los nodos de la red.

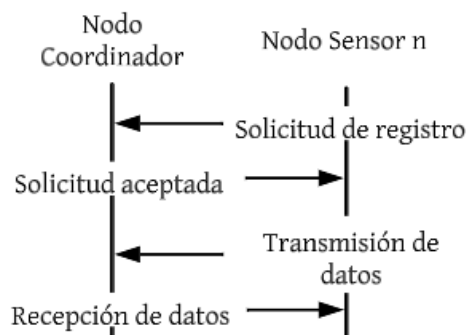


Nota. Dentro de la figura se puede observar que se representa en general la topología en estrella a simular.

De acuerdo a lo que se establece en la topología anterior la comunicación que se efectúa con cada uno de los nodos sensores, para que se identifiquen con el nodo coordinador y que finalmente la red se comunique de manera correcta.

Figura 10

Diagrama de secuencia de comunicación de los nodos sensores y el nodo coordinador.



Nota. En la figura se puede observar la secuencia de comunicación de los nodos de la red, en el cual el nodo sensor realiza una solicitud de registro para el nodo coordinador, donde este último responde con una solicitud aceptación. Fuente: Autoría.

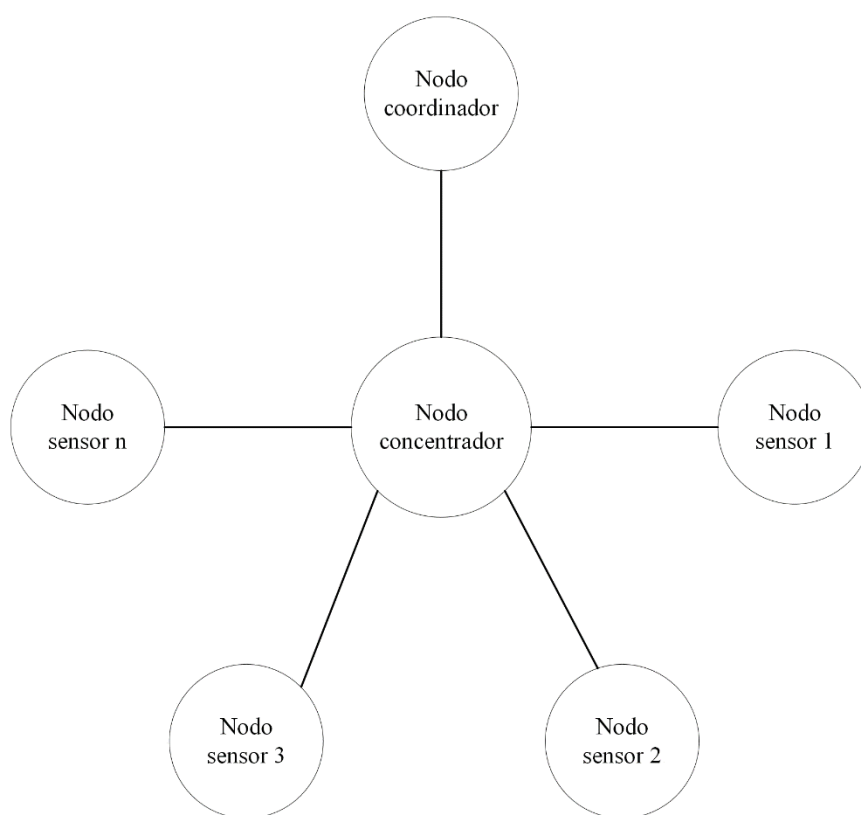
Una vez que se explica como sería la forma ideal de aplicar la topología a desarrollar, se debe tener en cuenta que el sistema a desarrollar requiere de la adición de otras herramientas para que sea completo, entre ellas se encuentra la integración del IDS Snort, pero según lo que se la topología para aplicarlo sería con la aplicación en cada uno de los enlaces del nodo sensor con el nodo coordinador, solución que no es factible debido al alto consumo de recursos que implicaría la puesta en acción de estos.

Por lo que, se opta por una solución en la que los nodos sensores se conecten con el nodo coordinador, siendo esta con la integración de un nodo concentrador el cual crea un enlace al nodo

coordinador, pero que en este enlace se encuentre la misma información que reciba el nodo concentrador, haciendo esto posible con la aplicación de la técnica que réplica el tráfico de la red. En la Figura 11 se observa lo detallado previamente, donde también se debe tener en cuenta que con esta solución permite que luego se realice su análisis y monitoreo.

Figura 11

Distribución de los nodos de la red con la solución para la integración del IDS.

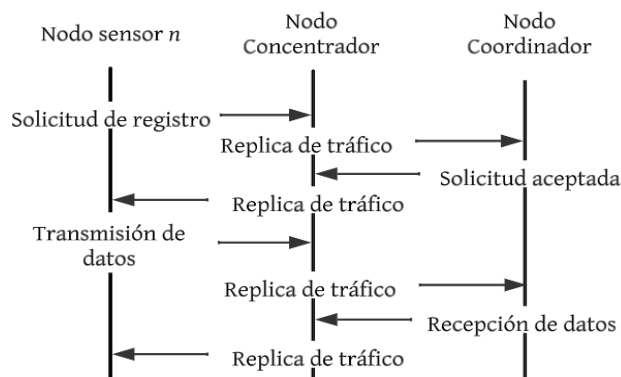


Nota. Según lo que se observa en la figura será posible la integración del IDS dentro de la topología debido al enlace de réplica de tráfico desde el nodo concentrador a nodo coordinador. Fuente: Autoría.

Teniendo la topología y saber cómo se encuentran enlazados se establece la comunicación del nodo sensor, nodo concentrador y nodo coordinador, donde se darán los procesos necesarios para que se establezca la transmisión entre estos de manera correcta.

Figura 12

Diagrama de secuencia de comunicación del nodo sensor, concentrador y coordinador.

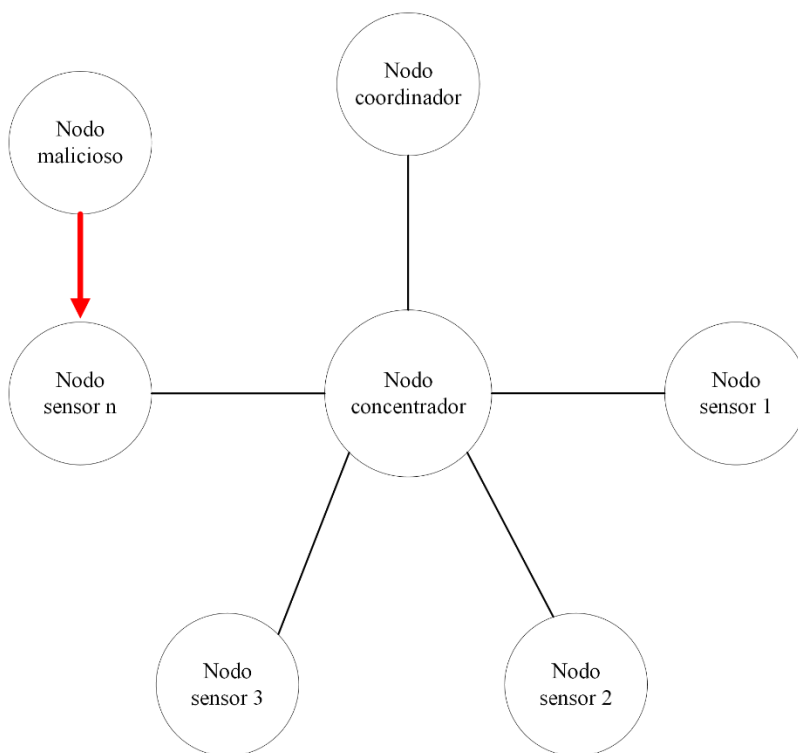


Nota. La secuencia de comunicación de los nodos de la red, en el cual el nodo sensor realiza una solicitud de registro para el nodo coordinador, pero dentro del proceso esta solicitud paso por primero por el nodo concentrador para finalmente llegar al nodo coordinador para que este último responde con una solicitud aceptación, pero primero llegando al nodo concentrador para finalmente llegar al nodo sensor. Fuente: Autoría.

Se debe resaltar que el diseño de la topología también cuenta con el nodo malicioso el cual proporcionara los ataques elegidos, permitiendo que después al integrar el IDS este logre enviar las alertas esperadas.

Figura 13

Distribución de los nodos de la red con la integración del nodo malicioso.

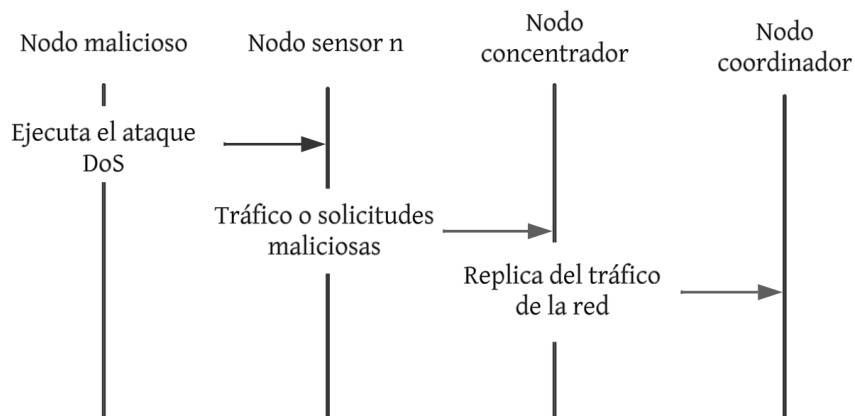


Nota. El diseño de la red también cuenta con el nodo malicioso que tiene como objetivo el ejecutar el ataque hacia la red. Fuente: Autoría.

Con la integración del nodo malicioso en la topología, se observa a continuación como sería la comunicación de los nodos y la influencia de este nuevo elemento.

Figura 14

Diagrama de secuencia de comunicación del nodo malicioso, nodo sensor, concentrador y coordinador.

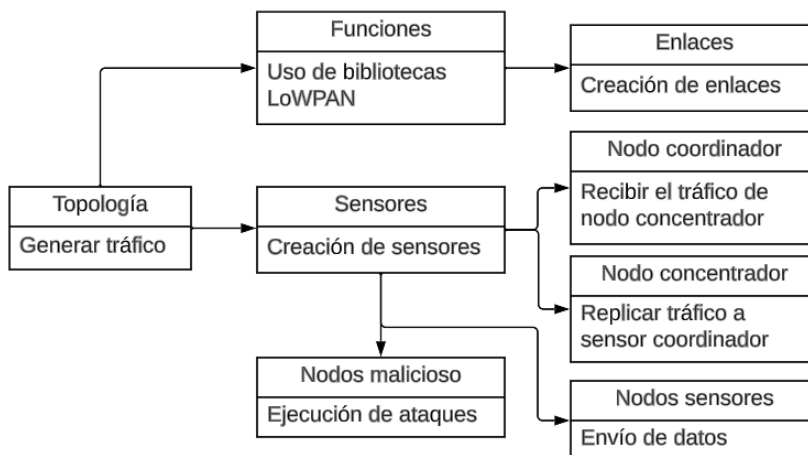


Nota. En este caso la comunicación dentro de la red es similar a los casos anteriores, pero la diferencia es que ahora dentro de esta replica de tráfico llegan junto con los ataques del nodo malicioso. Fuente: Autoría.

En la Figura 15 se detalla el diseño de la red dentro del software que se conforma de varios pasos, en los que se tiene en cuenta factores esenciales de la topología como la configuración de los nodos con sus respectivos parámetros, considerando que se puede adaptar a las necesidades del usuario, pero conociendo que existen limitaciones para el software en el que se trabaja, luego se configura los enlaces para que se efectúe la conexión de los nodos sensores con el nodo coordinador.

Figura 15

Diagrama del diseño de la red dentro del software.



Nota. El diagrama representa el diseño de la red dentro del software de simulación por lo que el bloque principal es la creación de la topología dentro del código para proceder con la creación de los nodos necesarios y la configuración de los enlaces de los mismo para su correcta comunicación.

Fuente: Autoría.

3.8. Integración del IDS

La integración del IDS dentro de la topología es de gran importancia, debido a que es una herramienta con la que se pueden realizar detecciones de anomalías dentro de la red, y permitiendo que emita alertas cuando se encuentran estos comportamientos sospechosos o maliciosos, por lo que, este IDS ayuda a que administradores de redes o personal de seguridad tome las medidas correspondientes para que la red no presente inconvenientes, promoviendo el monitoreo de los elementos restantes de la red para su funcionamiento óptimo.

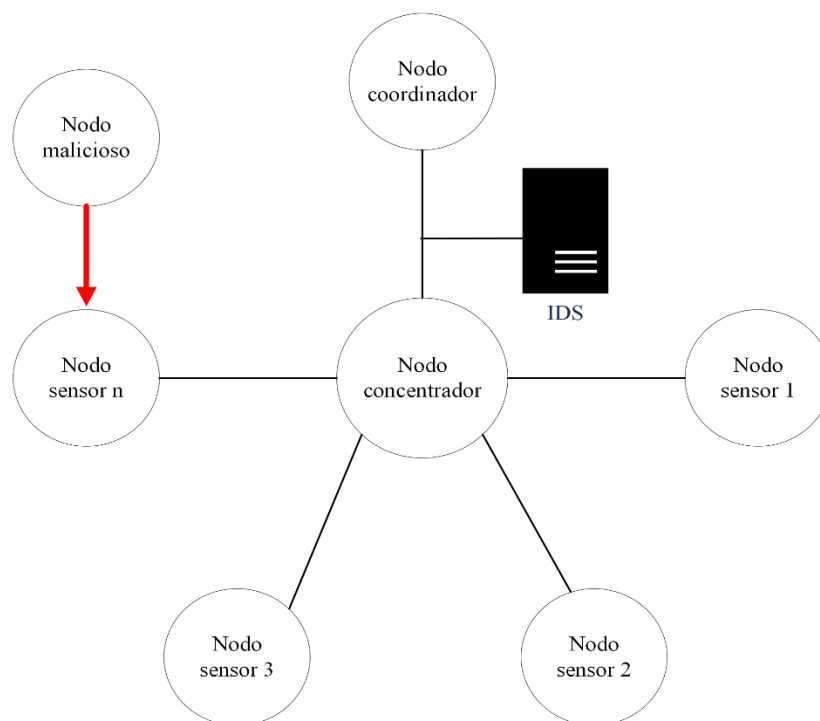
Sabiendo lo que un IDS puede realizar dentro de una red, al aplicarlo dentro de la WSN simulada, su ubicación es un punto esencial, por lo que, según lo detallado anteriormente la topología recurre a la técnica de la réplica del tráfico en un enlace hacia el nodo coordinador, la ubicación del IDS se aplica dentro de este, permitiendo que monitoree todos los datos recibidos y asegurándose de que no exista ningún tipo de comportamiento extraño que afecte a la red.

El prever el uso del IDS ubicado antes de un nodo coordinador se considera esencial, debido a que este es un importante de la red WSN, por lo que, este proceso representaría como un nivel de seguridad para posibles atacantes o comportamiento malicioso.

En la Figura 16 se puede observar cómo se realiza la integración del IDS dentro de la red. También se debe resaltar que la detección de ataques, intrusiones o comportamiento malicioso en la ubicación mencionada ayuda a que se obtenga una respuesta para contrarrestar de esta manera la amenaza antes de que esta llegue al nodo coordinador.

Figura 16

Integración del IDS dentro de la WSN.



Nota. Teniendo en cuenta lo explicado previamente se debe recordar que se realizaba una réplica de tráfico entre nodo concentrador y nodo coordinador, enlace que ahora se usa para la integración del IDS como se observa dentro de la figura. Fuente: Autoría.

3.8.1. Interconexión entre los diferentes componentes al IDS

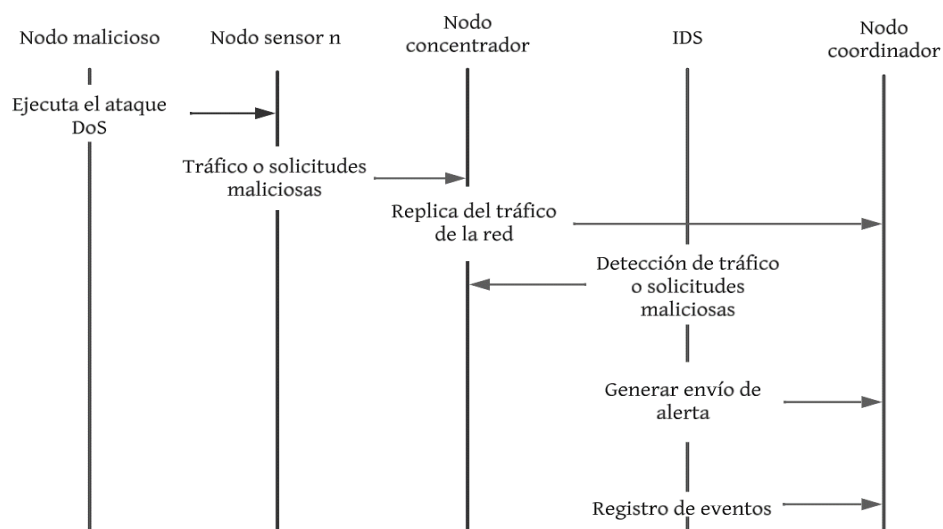
La integración del IDS para el entorno simulado que se realiza, tiene como objetivo que envíe alertas de un ataque que realiza un nodo malicioso dentro de la red, procurando obtener una alerta del ataque.

De acuerdo con la Figura 17 se puede establecer como se realiza la secuencia entre los distintos dispositivos que se encuentran involucrados a la hora de efectuar el ataque al nodo sensor de la

red. El proceso empieza cuando el nodo malicioso realiza el ataque DoS hacia la red la cual recibirá los paquetes maliciosos, generando la actividad inusual dentro de la red, para que el IDS lo detecte, registrando dicha actividad maliciosa y generando finalmente la alerta la cual se puede observar dentro del nodo coordinador.

Figura 17

Diagrama de secuencia del proceso que realizaría el nodo malicioso hacia la red, el efecto que tiene el IDS, envío de alerta y registro de eventos.

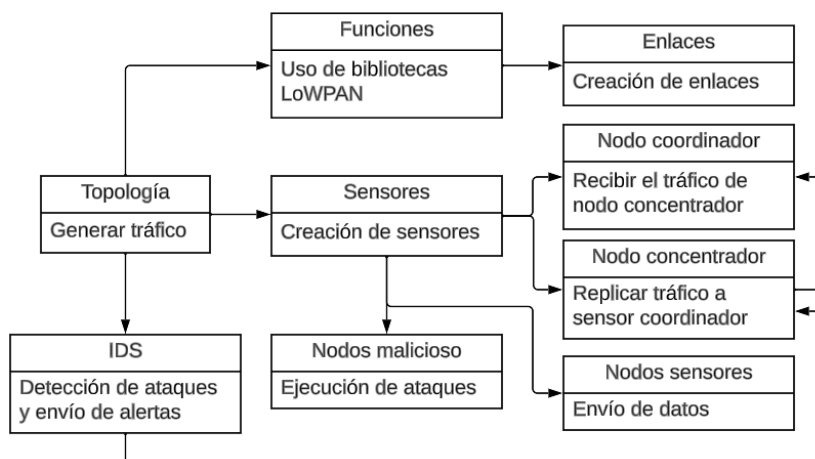


Nota. A diferencia de la Figura 14, en esta ya se encuentra en funcionamiento el IDS el cual se encuentra entre el nodo concentrador y nodo coordinador para la detección del tráfico malicioso y generar la alerta. Fuente: Autoría.

Continuando con el desarrollo de la red en el software se describe dentro de la Figura 18, observando que se agrega el IDS, inicie su ejecución y permitiendo de esta forma el monitoreo del tráfico de la red.

Figura 18

Diagrama del diseño de la red dentro del software.



Nota. Este diagrama representa la continuación de la Figura 15, en esta figura se puede observar que se agrega el IDS, el cual va a cumplir con sus objetivos de detectar los ataques y el envío de la alerta hacia el nodo concentrador. Fuente: Autoría.

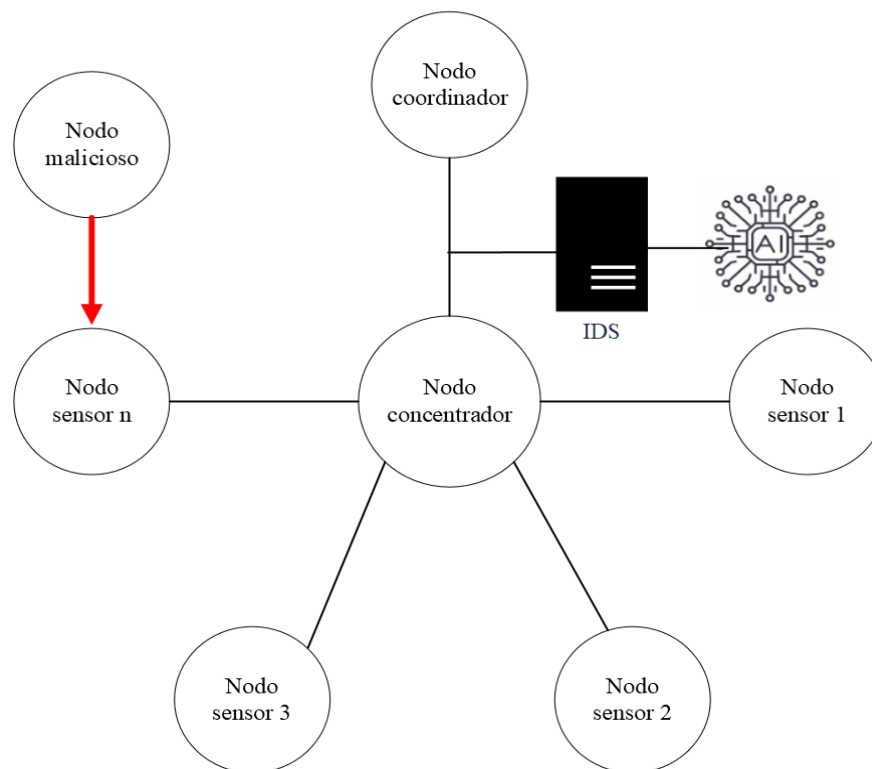
3.9. Integración de inteligencia artificial

En la fase 3 se desarrolla la WSN propuesta en un simulador, el cual permite implementar nodos y aplicar configuraciones que facilitan el despliegue del IDS, procedimientos de conexión que aseguran el funcionamiento del sistema y la verificación del comportamiento de los diferentes elementos de la red.

Esta sección se presenta la adición de la inteligencia artificial para que el sistema pueda detectar los comportamientos maliciosos y el envío de la alerta para proteger la red, en la Figura 19 se encuentra representado como se añade este elemento importante dentro de la construcción del sistema.

Figura 19

Topología de la red y la adición de la inteligencia artificial.



Nota. Esta figura representa el escenario completo del sistema, la implementación de los nodos (malicioso, sensores, concentrador y coordinador) e IDS con la integración de la Inteligencia Artificial. Fuente: Autoría.

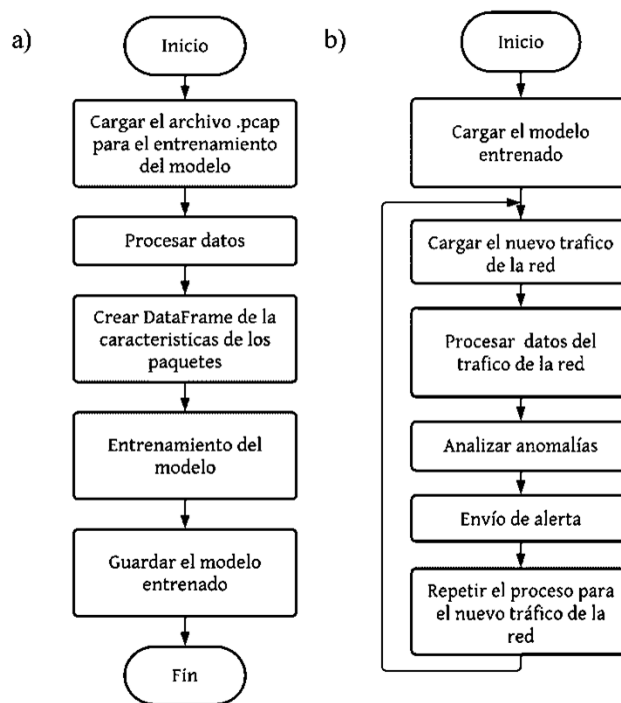
Para cumplir con el objetivo de aplicar la inteligencia artificial es el modelo de detección de anomalías Isolation Forest, este modelo tiene cierto parecido con el Random Forest donde se genera pequeños arboles de decisión, pero a diferencia de Isolation Forest este realiza unas particiones de manera aleatoria.

En la Figura 19 se observa el proceso del modelo de Isolation Forest que se puede ejecutar gracias a Python con Scikit-Learn o Sklearn librería que es utilizada para Machine Learning, la cual cuenta con algunas funciones que ayudan con estos procesos, como, por ejemplo, el transformar datos, creación de modelos supervisados y no supervisados, entre otros. En el caso aplicable del proyecto se realiza que el modelo capture en tiempo real los diferentes paquetes que se manejan dentro del tráfico de la red para establecer cuál de ellos es considerado como un ataque a la red, permitiendo así que envíe la alerta correspondiente. En este caso se considera como anomalía dentro de la red cuando el algoritmo identifica características que no concuerden con el comportamiento normal.

Se debe recalcar que este modelo es conocido por su eficacia en el manejo de amplios conjuntos de datos (Fei Tony Liu et al., 2008). Sin embargo, el que se utilicen múltiples muestras se generan desafíos para la identificación de las anomalías, debido a su naturaleza aleatoria se ve afectada por la abundancia de datos, lo que repercute en la capacidad para la detección de anomalías y puede que en otros casos se puedan generar falsos positivos.

Figura 20

Diagrama de flujo del funcionamiento del modelo Isolation Forest.

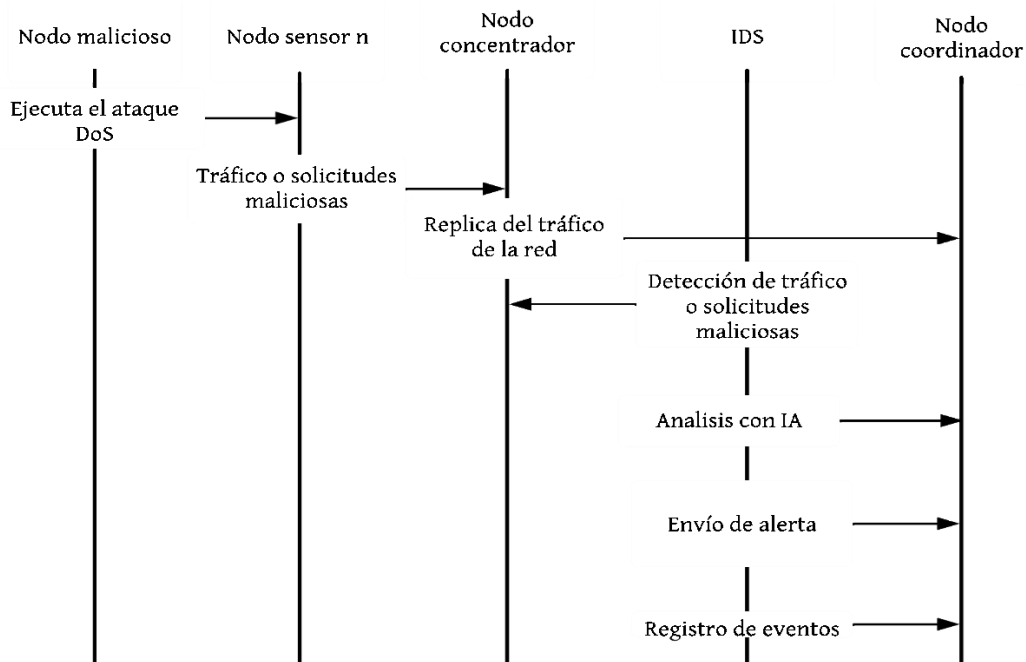


Nota. La integración del modelo Isolation Forest cuenta con dos fases las cuales permiten realizar: a) el entrenamiento del modelo y b) el análisis del nuevo tráfico de la red, este último no se encuentra con un proceso final debido a que este va a estar en constante funcionamiento hasta que se requiera detenerlo de manera manual. Fuente: Autoría.

En la Figura 21 se observa el diagrama de secuencia donde se añaden el proceso del modelo Isolation Forest, a diferencia de la Figura 17 que se presenta únicamente la integración del nodo malicioso e IDS.

Figura 21

Diagrama de secuencia donde se incluye la inteligencia artificial.



Nota. De acuerdo con la figura presentada, es el proceso final de la secuencia de la comunicación del sistema, debido a que, se incluye el funcionamiento del modelo Isolation Forest permitiendo que se generen las alertas tanto del IDS como del modelo implementado. Fuente: Autoría.

3.10. Funcionamiento de la configuración del sistema

Según las diversas secciones que se trataron dentro del capítulo donde se explica cada fase de cómo se desarrolla el sistema, ahora se explica el proceso donde se integra y se combinan todos los elementos, para que se pueda ejecutar el ataque a la WSN como de la detección del mismo, además del envío de la alerta.

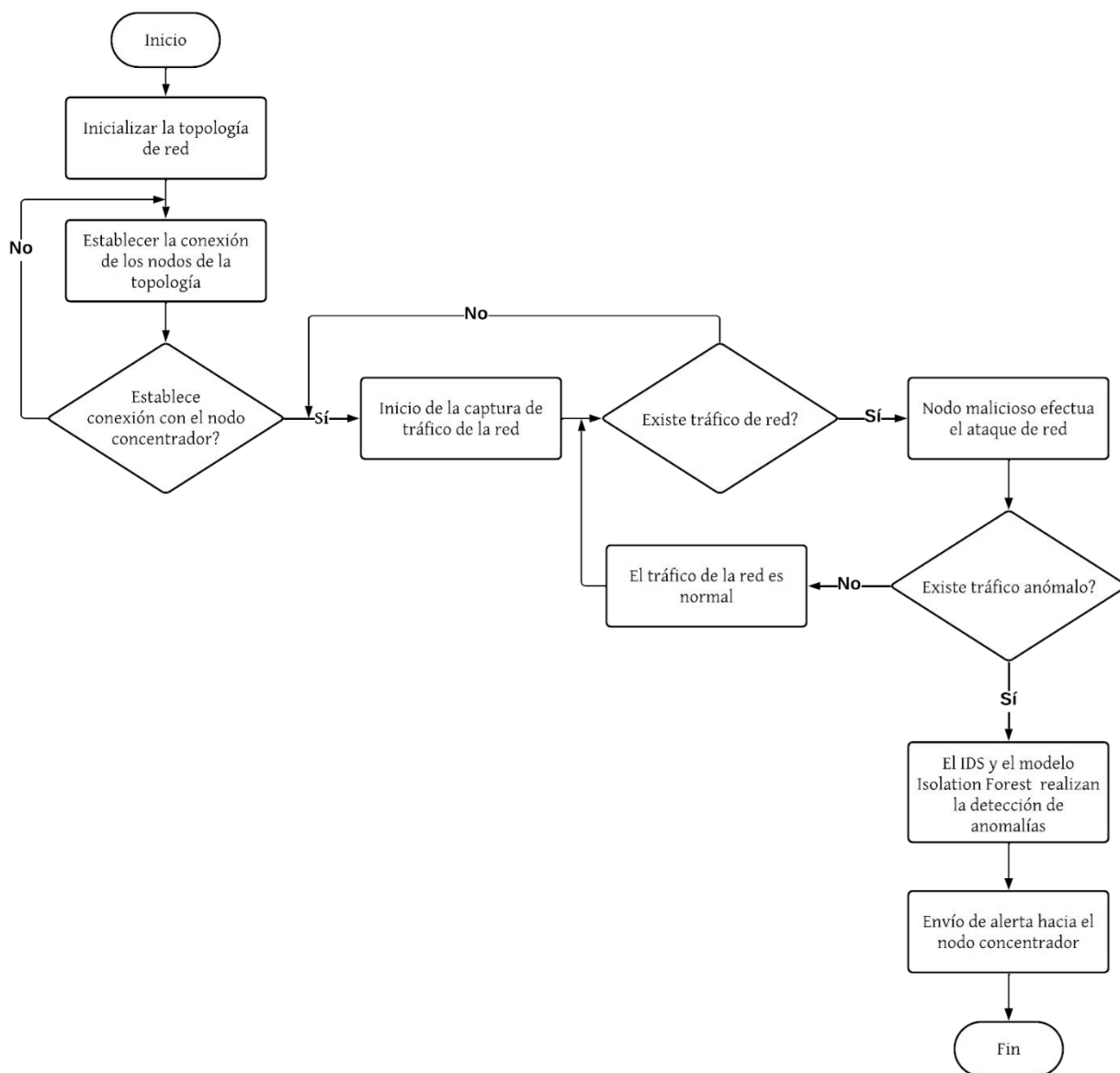
La configuración general de la red WSN de acuerdo con la detallado anteriormente se da con la creación de los nodos necesarios de la red, para después implementar el proceso de la detección

de intrusos y envío de alerta, usando el IDS y al modelo de Isolation Forest para que detecte los intrusos y genere la alerta.

La Figura 22 detalla el paso a paso de la ejecución de todo el proceso del sistema desarrollado, es decir, el ataque por parte del nodo malicioso, para que el IDS y el modelo Isolation Forest lo detecte, y emita las alertas por parte de los dos elementos. Si el tráfico continúa siendo normal, se seguirán capturando los paquetes a medida que la red se encuentre en ejecución. En caso de que se detecte anomalías, se procede a retomar el proceso previamente explicado.

Figura 22

Diagrama de flujo del funcionamiento completo del sistema.



Nota. La figura explica el proceso que sigue el sistema cuando se encuentra en funcionamiento y listo para realizar las pruebas necesarias. Fuente: Autoría.

CAPÍTULO IV IMPLEMETACIÓN Y PRUEBAS DE FUNCIONAMIENTO

Este capítulo trata de las pruebas de funcionamiento del IDS en la red WSN diseñada utilizando técnicas de inteligencia artificial. Además de describir los resultados obtenidos durante su evaluación al entorno sometido. Evaluando la eficacia del IDS en la detección y prevención de posibles intrusiones en la red y analizar cómo el uso de técnicas de IA puede mejorar la capacidad de detección y reducir el número de falsas alarmas.

Siendo así aplicable la fase 5 de la metodología aplicada, donde se exponen las pruebas para monitorizar el comportamiento del IDS instalado en la red. Estas pruebas pueden producir resultados tales como el envío de notificaciones de que la red ha sido atacada y comprometida.

4.1. Implementación

El diseño de la red se basa en la configuración de los nodos sensores, nodo concentrador y nodo coordinador, los nodos sensores en general usan líneas de código específicos para nombrar cada nodo, la dirección IPv6 asignada al sensor, el identificador de la red personal de área de servicio (PAN ID), modo de almacenamiento del sensor en el caso de la topología implementada se usa uno de tipo 0 debido a que no se necesita que se guarde información de enrutamiento, posición para la gráfica del nodo, y la especificación de la interfaz que se usa para la conexión de los sensores a la red simulada, se debe resaltar que para el nodo coordinador y concentrador se usa un comando adicional donde se establece que actúa como la raíz de DODAG (Grafo Acíclico Dirigido Orientado a Destino) estructura usada para redes basadas en RPL (Protocolo de Enrutamiento para Redes de Baja Potencia y Perdida) características que se encuentran comúnmente en ambientes de redes de sensores e IoT.

En la Figura 23 se observan las líneas de código descritas anteriormente, detallando la configuración para cada que se usa dentro de la simulación.

Figura 23

Creación de los nodos de la topología WSN.

```

info("*** Creación de nodos\n")
# NODO_CONCENTRADOR
sensor1 = net.addSensor('sensor1', ip6='fe80::1/64', panid='0xbeef', dodag_root=True,
storing_mode=0, position='125,100,0', intf='wlp2s0')
# NODO_COORDINADOR
sensor2 = net.addSensor('sensor2', ip6='fe80::2/64', panid='0xbeef', dodag_root=True,
storing_mode=0, position='125,200,0', intf='wlp2s0')
# NODO_SENSOR1
sensor3 = net.addSensor('sensor3', ip6='fe80::3/64', panid='0xbeef', storing_mode=0,
position='200,150,0', intf='wlp2s0')
# NODO_SENSOR2
sensor4 = net.addSensor('sensor4', ip6='fe80::4/64', panid='0xbeef', storing_mode=0,
position='200,50,0', intf='wlp2s0')
# NODO_SENSOR3
sensor5 = net.addSensor('sensor5', ip6='fe80::5/64', panid='0xbeef', storing_mode=0,
position='50,50,0', intf='wlp2s0')
# NODO_SENSOR4
sensor6 = net.addSensor('sensor6', ip6='fe80::6/64', panid='0xbeef', storing_mode=0,
position='50,150,0', intf='wlp2s0')
# NODO_MALICIOSO
sensor7 = net.addSensor('sensor7', ip6='fe80::75df:5d28:47bf:7f6b/64', panid='0xbeef',
storing_mode=0, position='175,200,0', intf='wlp2s0')

```

Nota. Dentro de la configuración de los nodos concentrador y coordinador se observa el comando `dodag_root` con el que se diferencian de los demás nodos sensores dentro de la simulación. Fuente: Autoría.

Para continuar con la configuración de la red se establecen los enlaces de los nodos sensores donde se cumple con la topología en estrella, en otras palabras, los nodos sensores establecen la conexión solo con el nodo concentrador y únicamente el nodo concentrador se comunica con el nodo coordinador, considerar como se encuentran distribuidos los nodos según la Figura 23.

Dentro de la Figura 24 se agrega el modelo de propagación *logDistance* que se establece dentro del entorno de simulación de acuerdo con el análisis realizado previamente donde se establece que este modelo puede ser aplicado en entornos de exteriores donde existen o no obstáculos con los que se vea afectada la comunicación entre sensores, por tanto, se establece el exponente el cual

influye en la rapidez con la que la intensidad de la señal de radio disminuye con la distancia (mientras más alto el valor del exponente, la caída de la intensidad con la distancia es más rápida).

Figura 24

Configuración de los enlaces de los nodos sensores y modelo de propagación.

```
info("*** Configuración de nodos\n")
net.configureNodes()

info("*** Creando enlaces\n")
net.addLink(sensor1, sensor3, cls=LoWPAN)
net.addLink(sensor1, sensor4, cls=LoWPAN)
net.addLink(sensor1, sensor5, cls=LoWPAN)
net.addLink(sensor1, sensor6, cls=LoWPAN)
net.addLink(sensor2, sensor1, cls=LoWPAN)
net.addLink(sensor7, sensor1, cls=LoWPAN)
net.addLink(sensor7, sensor3, cls=LoWPAN)
net.addLink(sensor7, sensor4, cls=LoWPAN)
net.addLink(sensor7, sensor5, cls=LoWPAN)
net.addLink(sensor7, sensor6, cls=LoWPAN)

info("*** Configuración de propagación del modelo\n")
net.setPropagationModel(model="logDistance", exp=4)
```

Nota. La creación de los enlaces de cada nodo sensor se basa en que cada uno de los nodos sensores se conecte hacia el nodo concentrador y que este último se conecte únicamente hacia el nodo coordinador, por otro lado, el nodo atacante en este caso tendrá acceso a los nodos sensores y al nodo concentrador. Fuente: Autoría.

Por otra parte, el generar la lectura de la temperatura de los nodos sensores y el envío de la lectura realizada se basa en tres métodos con los que se complementan. En la primera función se genera la temperatura entre un rango de 20 a 30 grados Celsius, para luego en la segunda función genere el paquete y enviarlo al destino, y en la función final se establece el tiempo en que se requiere que se envíe cada una de las lecturas sin antes realizar la lista de los sensores de envío y destino, además de que hace que este proceso se repita mientras la simulación se encuentre en ejecución.

La Figura 25 muestra cómo se encuentra estructurado el envío de la lectura de temperatura detallada anteriormente, mensajes que se observan dentro de la simulación asegurando que se envió y especificando a que nodo corresponde.

Figura 25

Configuración de la generación de la lectura de temperatura y envío de la lectura de la simulación.

```
def generate_random_temperature():
    return random.uniform(20.0, 30.0)

def generate_and_send_temperature(sensor, destination):
    src_ip = sensor.IP()
    dst_ip = destination.IP()

    temperature = generate_random_temperature()

    icmpv6_packet = IPv6(src=src_ip, dst=dst_ip)/ICMPv6EchoRequest()/
Raw(load=f'Temperature: {temperature}')

    send(icmpv6_packet, iface='wlp2s0')

    info(f"Sensor {sensor.name} envía lectura de temperatura:
{temperature}\n")
    sensor.cmd(f"echo {temperature} | nc -6u -w1 {destination.IP()}
12345")

def generate_traffic(net):
    sensors = [net.getNodeByName(f"sensor{i}") for i in range(3, 7)]
    sensor1 = net.getNodeByName("sensor1")

    while True:
        for sensor in sensors:
            generate_and_send_temperature(sensor, sensor1)
```

Nota. Las funciones del código muestran la generación de tráfico de temperatura de los nodos de la red simulada. Fuente: Autoría.

Continuando con la configuración de la red, se establece una parte importante en la que se realiza la réplica de tráfico, capturado en la interfaz del nodo concentrador y guardando su captura en segundo plano, para que ahora el nodo coordinador reproduzca el tráfico capturado.

En la Figura 26 se observa el proceso para su configuración donde la ejecución de los comandos para el uso del IDS dentro de la simulación y permitir que funcione de la manera correcta.

Figura 26

Configuración de la réplica del tráfico de la red.

```
info("*** Captura de tráfico en el sensor1\n")
capture_command = "tcpdump -i {} -w /tmp/capture.pcap &".format(sensor1.wintfs[0].name)
sensor1.cmd(capture_command)

time.sleep(10)

info("*** Réplica del tráfico en el sensor2\n")
replay_command = "tcpreplay -i {} /tmp/capture.pcap".format(sensor2.wintfs[0].name)
sensor2.cmd(replay_command)
```

Nota. La réplica del tráfico se debe realizar dentro de la función de la topología para que este se ejecute en conjunto con la inicialización de los nodos sensores. Fuente: Autoría.

La fase 4 establece una comunicación adecuada entre la red y el sistema, los procedimientos mediante los cuales se gestionará la implantación del sistema global e identificar los posibles problemas que podrían impedir que el IDS o sus componentes individuales cumplieran la finalidad prevista.

La integración del IDS dentro de la red WSN también se da dentro del archivo de la configuración de cada una de las funciones que se necesitan para la inicialización de la red.

En la Figura 27 se muestra cómo se agrega el nodo para el IDS, su nombre como dispositivos de la red, la dirección IPv6 y el enlace hacia al nodo concentrador, para que pueda realizar la captura del tráfico de la red, se debe tener en cuenta que se agrega la inicialización del IDS desde la ruta donde se encuentra instalado, además de que se debe detener el uso del IDS cuando se termina la ejecución de la toda la topología.

Figura 27

Agregar el nodo IDS usando Snort.

```

info("*** Agregando nodo IDS \n")
ids = net.addHost('ids', ip6='fe80::8/64')

info("*** Creando enlace entre el IDS y sensor principal\n")
net.addLink(ids, sensor1)

info("*** Iniciando IDS (Snort)\n")
ids.cmdPrint('/usr/sbin/snort -A console -q -u snort -g snort -c /etc/snort/snort.conf -i
sensor1-wpan0 &')

info("*** Ejecución de CLI\n")
CLI(net)

```

Nota. De acuerdo con la figura se puede observar que el IDS se agrega el enlace con respecto al nodo concentrador, pero se debe considerar que se realiza la réplica del tráfico hacia el nodo coordinador, por lo que, al ubicarlo de esta manera es mucho más rápida su detección. Fuente: Autoría.

A pesar de que se integre el IDS dentro de la configuración de la red, se debe tener en cuenta que se hacen varias configuraciones dentro de archivos pertenecientes a Snort, es decir que además de incluir el nodo y la ubicación de donde se encuentra instalado, se espera cambios en el archivo principal de snort.conf donde se puede manipular para poder agregar la red a la que se requiere analizar o simplemente usarla de manera predeterminada. Los componentes del IDS representan la configuración del archivo principal de Snort donde se agregan parámetros y aspectos para que funcione correctamente.

Figura 28

Edición del archivo general para el funcionamiento de Snort.

```

ipvar HOME_NET any
# Set up the external network addresses. Leave as "any" in most situations
ipvar EXTERNAL_NET any
var EXTERNAL_NET any
# If HOME_NET is defined as something other than "any", alternative, you can
# use this definition if you do not want to detect attacks from your internal
# IP addresses:
#ipvar EXTERNAL_NET !$HOME_NET

```

Nota. El archivo se puede modificar según los requerimientos establecidos, así que se pueden establecer configuraciones adicionales según las necesidades. Fuente: Autoría.

Para que el IDS utilizado funcione de la manera que se requiere se deben establecer las alertas a utilizar para la detección del ataque, a continuación, se muestra la Tabla 13 donde se indica como se encuentra compuesta cada una de ellas.

Tabla 13

Configuración de alertas para la detección de ataques.

	ALERTA 1	ALERTA 2
Tipo de alerta	alert icmp	alert icmp
Origen y destino	any any -> any any	any any -> any any
Mensaje	msg:"Ataque Flood_RS6 (RS)"	msg:"Ataque Flood_Redir6 (RE)"
Tipo de ICMP	itype:133	itype:137
Código ICMP	icode:0	icode:0
Filtro de detección	detection_filter:track by_dst, count 50000, seconds 20	detection_filter:track by_dst, count 50000, seconds 20
Clasificación de política	classtype:policy-violation	classtype:policy-violation
Identificador único	sid:10000021	sid:10000022
Número de revisión	rev:3	rev:3

Nota. El tipo de tráfico en este caso es ICMP, donde se especifica el origen y destino del tráfico, para cada una de las reglas se agrega el mensaje correspondiente para el ataque en específico, para

los tipo de mensaje ICMP es de acuerdo a lo requerido, en este caso para 133 se asocia con mensajes de RS (Router Solicitation) y para 137 para RE (Redirect), para el código específico de ICMP es 0, el tipo de clasificación de la alerta es de violación de política, el sid representa la identificación única para la regla, y rev para el numero de las revisiones de la regla.

Finalmente se configura el modelo Isolation Forest el cual ayuda también a la detección de anomalías dentro del tráfico de la red, considerando que se este se encuentra adaptado para que trabaje a la par con el IDS.

El funcionamiento del modelo comienza con el previo entrenamiento de este, cargando una captura de paquetes de la red, usado para la extracción de la información más relevante de los paquetes de la red, guardando el modelo entrenado para posteriormente la detección de anomalías de los nuevos datos de la red. En la Figura 29 las funciones que se usan para el entrenamiento se añade la ruta del archivo de entrada, datos que se leerán gracias a las librerías Scapy para crear información relevante, se observa que se agrega una instancia del modelo donde se agrega el número máximo de muestras mientras más muestras es más robusto y preciso, también se establece el número máximo de muestras para extraer y entrenar el modelo (cuando se le asigna “auto” se seleccionara un número automático adecuado de las muestras), la parte de contaminación es el porcentaje aproximado para que los datos sean anómalos.

Figura 29

Configuración del archivo de entrenamiento del modelo.

```

archivo_pcap = '/home/jhose/Documentos/mininet/mininet-wifi/examples/WSN/
entrenamiento0.pcap'
def cargar_y_procesar_datos(archivo_pcap):
    packets = rdpcap(archivo_pcap)

    data = pd.DataFrame({
        'protocolo': [packet.strftime('%IP.proto%') if IP in packet else 'Non-IP' for
        packet in packets],
        'tamano_paquete': [len(packet) for packet in packets],
        'puerto_origen': [packet.sport if TCP in packet else None for packet in
        packets],
        'puerto_destino': [packet.dport if TCP in packet else None for packet in
        packets],
    })

    data = pd.get_dummies(data, columns=['protocolo'], prefix=['protocolo'])
    data = data.fillna(0)

    return data

data = cargar_y_procesar_datos(archivo_pcap)

modelo_isof = IsolationForest(
    n_estimators=1000,
    max_samples='auto',
    contamination=0.01,
    n_jobs=-1,
    random_state=123
)

modelo_isof.fit(X=data)

import joblib
joblib.dump(modelo_isof, 'modelo_isof.pkl')

```

Nota. Al configurar las funciones necesarias, se observa el método fit con el que ahora se puede realizar la detección de anomalías en los nuevos datos capturados por la red. Fuente: Autoría.

Como se menciona anteriormente, primero se realizó el entrenamiento del modelo, por lo que, ahora se procede con el análisis de los nuevos datos de la red, configurando la alerta si se encuentran las anomalías y quedando en un bucle para que cargue y procese los datos del archivo, además considerando que el monitoreo se realiza en tiempo real.

La Figura 30 muestra cómo se encuentran estructuradas cada una de las funciones que se usan para poder enviar la respectiva alerta de anomalía dentro de la red.

Figura 30

Configuración del archivo de análisis de la captura de paquetes de la red con el modelo entrenado.

```

archivo_pcap = '/home/jhose/Documentos/mininet/mininet-wifi/examples/WSN/
sensor_coordinator.pcap'
modelo_isof = joblib.load('modelo_isof.pkl')

def cargar_y_procesar_datos(archivo_pcap):
    packets = rdpcap(archivo_pcap)
    data = pd.DataFrame({
        'protocolo': [packet.strftime('%IP.proto%') if IP in packet else 'Non-IP'
        packet in packets],
        'tamaño_paquete': [len(packet) for packet in packets],
        'puerto_origen': [packet.sport if TCP in packet else None for packet in
        packets],
        'puerto_destino': [packet.dport if TCP in packet else None for packet in
        packets],
    })

    data = pd.get_dummies(data, columns=['protocolo'], prefix=['protocolo'])
    protocolo_columnas = ['protocolo_TCP', 'protocolo_UDP', 'protocolo_Non-IP']
    for columna in protocolo_columnas:
        if columna not in data.columns:
            data[columna] = 0

    data = data.fillna(0)
    return data

def analizar_anomalias(data):
    columnas_dummy_actuales = data.columns.tolist()
    modelo_isof.columns_ = columnas_dummy_actuales

    if modelo_isof.n_features_ != len(columnas_dummy_actuales):
        modelo_isof.n_features_ = len(columnas_dummy_actuales)

    clasificacion_predicha = modelo_isof.predict(X=data)
    score_anomalia = modelo_isof.score_samples(X=data)
    cuantil_01 = np.quantile(score_anomalia, q=0.01)

    if -1 in clasificacion_predicha:
        print("¡Alerta! Se encontraron anomalías en el tráfico de la red.")

```

Nota. El código permite el monitoreo de manera continua y detecta las anomalías para luego emitir la alerta. Fuente: Autoría.

Una vez configurado tanto la simulación de la topología, como la configuración del IDS y el modelo de inteligencia artificial seleccionado, se procede a realizar las respectivas pruebas con las que se comprueba el funcionamiento de todas estas.

4.2. Pruebas de funcionamiento

Para comprobar el funcionamiento de la topología y las herramientas adicionales que se aplican para completar el sistema, se ejecuta el ataque DoS para comprobar el envío de alertas mediante

cada una de las utilidades que se establecieron previamente. Permitiendo así verificar el correcto funcionamiento del sistema desarrollado.

4.2.1. *Plan de pruebas*

El plan de pruebas se realiza con el fin de garantizar la efectividad de las pruebas para cada uno de los parámetros que se encontraban predefinidos en secciones pasadas, además de permitir que se verifique el desempeño del sistema simulado, este proceso se observa a continuación:

Tabla 14

Plan de pruebas.

PLAN DE PRUEBAS		
Tipo de prueba	Descripción	Resultado
Pruebas básicas	Evidenciar las funciones principales del sistema.	Conectividad de la red Notificación del IDS Notificación del modelo Isolation Forest
Pruebas específicas	Evidenciar aspectos particulares del sistema	Ejecución de ataques Funcionalidad completa del sistema Pruebas de tráfico normal Pruebas de tráfico anómalo Detección de ataques

Nota. El tipo de pruebas que se establecen dentro de la tabla sirven para poder evidenciar el funcionamiento del sistema desarrollado.

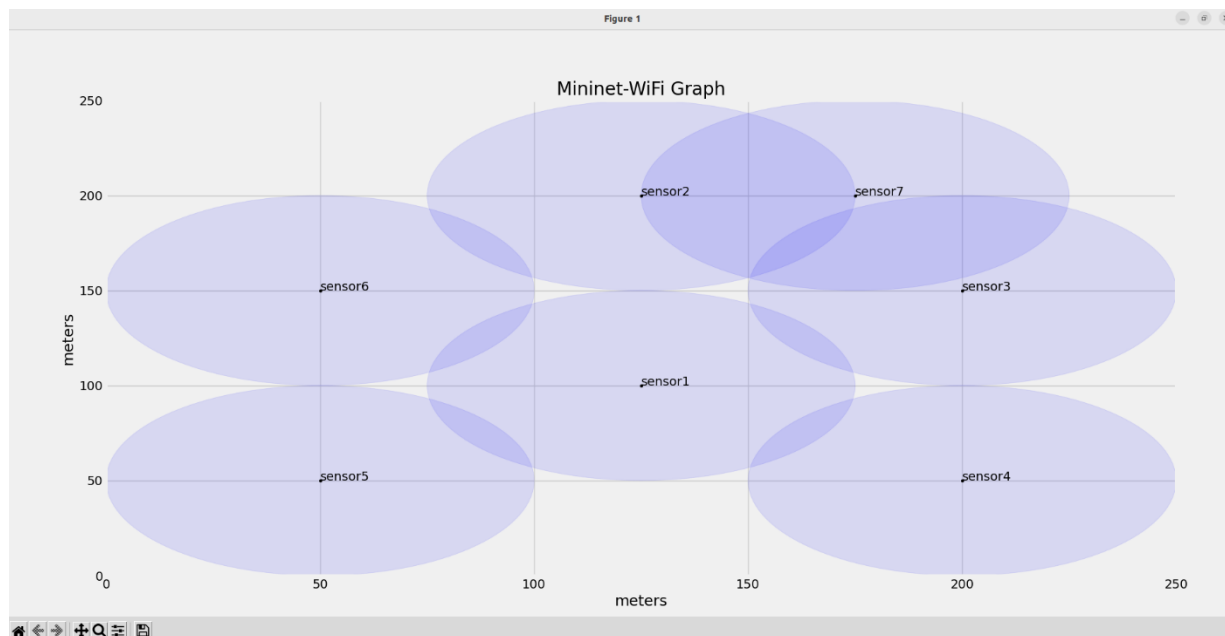
4.2.2. *Pruebas básicas*

Como se establece dentro del plan de pruebas esta sección tiene como objetivo que las pruebas a aplicar se enfoquen en garantizar las funciones principales del sistema, es decir que operen de acuerdo con los parámetros que se establecieron en capítulos anteriores.

Una vez que se configura lo requerido, se puede ejecutar los archivos, donde lo primero que se puede observar que se obtiene es la gráfica de la red tal como se muestra en la Figura 31.

Figura 31

Gráfica de la red de sensores inalámbricos.



Nota. La figura muestra cómo se encuentran distribuidos los diferentes nodos que se configuran dentro de la red en Mininet-wifi. Fuente: Autoría.

La demostración de la conectividad de la red se puede comprobar en la Figura 32, en la que se realiza un ping desde cada uno de los nodos sensores y el nodo concentrador.

Figura 32

Comprobación de conectividad de la red.

```

mininet-wifi> sensor1 ping6 -c1 fe80::6
PING fe80::6(fe80::6) 56 data bytes
64 bytes from fe80::6%sensor1-pan0: icmp_seq=1 ttl=64 time=0.402 ms

--- fe80::6 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.402/0.402/0.402/0.000 ms
mininet-wifi> sensor1 ping6 -c1 fe80::5
PING fe80::5(fe80::5) 56 data bytes
64 bytes from fe80::5%sensor1-pan0: icmp_seq=1 ttl=64 time=0.280 ms

--- fe80::5 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.280/0.280/0.280/0.000 ms
mininet-wifi> sensor1 ping6 -c1 fe80::4
PING fe80::4(fe80::4) 56 data bytes
64 bytes from fe80::4%sensor1-pan0: icmp_seq=1 ttl=64 time=0.357 ms

--- fe80::4 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.357/0.357/0.357/0.000 ms
mininet-wifi> sensor1 ping6 -c1 fe80::3
PING fe80::3(fe80::3) 56 data bytes
64 bytes from fe80::3%sensor1-pan0: icmp_seq=1 ttl=64 time=0.329 ms

--- fe80::3 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.329/0.329/0.329/0.000 ms
mininet-wifi> █

```

Nota. La comprobación se realiza con cada uno de los sensores correspondientes de la red.

Fuente: Autoría.

De acuerdo con las utilidades implementadas dentro del sistema es posible observar cómo Snort muestra las alertas configuradas mientras se encuentra en ejecución la topología. La ventana que se muestra en la Figura 33 es posible obtener ejecutando el comando a continuación **sudo snort -A console -c /etc/snort/snort.conf -i <interfaz> -N**.

Figura 33

Salida de alertas de detección de Snort.

```

12/04-02:14:59.520108 100000022:3] ATAQUE FLOOD REDIR6 (RE) CON THCIpV
6 6 [**] [Classification: Potential Corporate Privacy Violation] [Priority: 1] {
IPV6-ICMP} fe80::75df:5d28:47bf:7f6b -> ff02::1
12/04-02:14:59.520112 100000022:3] ATAQUE FLOOD REDIR6 (RE) CON THCIpV
6 6 [**] [Classification: Potential Corporate Privacy Violation] [Priority: 1] {
IPV6-ICMP} fe80::75df:5d28:47bf:7f6b -> ff02::1
12/04-02:14:59.520117 100000022:3] ATAQUE FLOOD REDIR6 (RE) CON THCIpV
6 6 [**] [Classification: Potential Corporate Privacy Violation] [Priority: 1] {
IPV6-ICMP} fe80::75df:5d28:47bf:7f6b -> ff02::1
12/04-02:14:59.520122 100000022:3] ATAQUE FLOOD REDIR6 (RE) CON THCIpV
6 6 [**] [Classification: Potential Corporate Privacy Violation] [Priority: 1] {

```

Nota. La salida que muestra el IDS es el tipo de protocolo que se está detectando, el mensaje que se añade según corresponda y la clasificación de la violación de la política. Fuente: Autoría.

La Tabla 15 que se detalla a continuación muestra los parámetros que se muestra en la salida de Snort:

Tabla 15

Salida de alertas de la detección con el IDS.

	ALERTA 1	ALERTA 2
Fecha y hora	12/04-01:45:34.622140	12/04-03:14:58.708673
Identificador único	[1:10000021:3]	[1:10000022:3]
Mensaje de la alerta	Ataque Flood_Rs6 (RS)	Ataque Flood_Redir6 (RE)
Clasificación de la alerta	[Classification: Potential Corporate Privacy Violation]	[Classification: Potential Corporate Privacy Violation]
Prioridad	[Priority: 1]	[Priority: 1]
	{IPV6-ICMP}	{IPV6-ICMP}
Conexión	fe80::75df:5d28:47be:6fd -> ff02::1	fe80::75df:5d28:47be:6fd -> ff02::1

Nota. En la tabla se muestra la alerta detectada con el IDS, se debe considerar que estas pueden ser modificadas según la necesidad del ataque que se quiere detectar.

Al encontrarse en ejecución tanto la topología como la ventana de las salidas de alertas de Snort se puede ejecutar el modelo Isolation Forest. Siendo así que en la Figura 34 se muestra la salida de consola en donde indica que existe una anomalía en la red y por la espera de nuevos paquetes para ser analizado.

Figura 34

Salida de alertas de detección de anomalías del modelo Isolation Forest.

```
¡Alerta! Se encontraron anomalías en el tráfico de la red.  
Esperando para la próxima iteración..  
¡Alerta! Se encontraron anomalías en el tráfico de la red.  
Esperando para la próxima iteración..  
¡Alerta! Se encontraron anomalías en el tráfico de la red.  
Esperando para la próxima iteración..  
¡Alerta! Se encontraron anomalías en el tráfico de la red.
```

Nota. Una vez ejecutado el código del modelo se observa la alerta emitida de que se encontró la anomalía dentro de la red y a la espera de una nueva iteración según la configuración realizada.

Fuente: Autoría.

Una vez que se encuentra comprobado el funcionamiento de todo el sistema se procede con la ejecución del ataque detallado a continuación:

La Figura 35 presenta al nodo malicioso que ejecuta el primer ataque DoS, como se observa se efectúa el ataque **flood_rs6 -s** que tiene como objetivo inundar de paquetes a la red con mensajes ICMPv6 de solicitud de enrutador, flood hace que los paquetes se envíen tan rápido como sea posible sin presentar intervalos.

Figura 35

Ataque de inundación de mensajes ICMPv6.

```
jhose@PORTEGE-Z30t-C:~/Documentos/mininet/mininet-wifi/examples/WSN$ sudo atk6-flood_rs
6 -s wlp2s0
Starting to flood with ICMPv6 router solicitation on wlp2s0 (Press Control-C to end, a
dot is printed for every 1000 packets):
.....^C
jhose@PORTEGE-Z30t-C:~/Documentos/mininet/mininet-wifi/examples/WSN$
```

Nota. El ataque DoS que se ejecuta dentro de la simulación inunda la red con mensajes ICMPv6 de solicitudes de enrutador. Fuente: Autoría.

De igual manera la Figura 36 representa el segundo ataque DoS, se ejecuta **flood_redir6 -F** el cual envía una gran cantidad de paquetes a un objetivo con redirecciones ICMPv6, cumpliendo con el objetivo de que se sobrecargue el servicio.

Figura 36

Ataque de inundación de redirecciones ICMPv6.

```
jhose@PORTEGE-Z30t-C:~/Documentos/mininet/mininet-wifi/examples/WSN$ sudo atk6-flood_re
dir6 -F wlp2s0
Starting to flood with ICMPv6 redirects on wlp2s0 (Press Control-C to end, a dot is pri
nted for every 1000 packets):
.....^C
jhose@PORTEGE-Z30t-C:~/Documentos/mininet/mininet-wifi/examples/WSN$
```

Nota. El ataque DoS que se ejecuta dentro de la simulación inunda un objetivo con redirecciones ICMPv6. Fuente: Autoría.

4.2.3. Pruebas específicas

Las pruebas específicas tienen el objetivo de evidenciar los aspectos particulares del sistema, las cuales superan las funcionalidades básicas, estas se enfocarán tanto el tráfico normal como el anómalo, además de la detección de anomalías dentro de la red.

Se debe tener en cuenta que las pruebas se basan en casos reales donde establecen que este tipo de redes se encuentran en estados inactivos y luego operativos para la transmisión de datos. La mayor parte del tiempo estos nodos permanecen en un modo llamado durmiente. Cuando se realizan los análisis correspondientes dentro de este estudio real menciona que estos se pueden activar en periodos breves como por ejemplo en un periodo de 30 segundos y en otros casos es más prolongado en un estimado de 2 minutos y 10 segundos, resultados que nos dan una visión a los patrones que usan este tipo de redes WSN (Muñoz Barragán Sócrates Nelson et al., 2017).

Al juntar los procesos a tiempo real dentro de la topología se observa la Figura 37 mostrando la salida de las alertas del IDS Snort, como las de Isolation Forest se reciben de manera correcta a la hora de ejecutar el ataque previamente explicado. Teniendo en cuenta que la ventana del IDS Snort se encontrara en una constante actualización según como detecte los ataques que se implementen, mientras que la ventana de Isolation Forest continua con el envío de la alerta según lo vaya detectando y a la espera de nuevos paquetes de acuerdo con el tráfico existe dentro de la red, comprobando de esta manera la funcionalidad del sistema en su totalidad.

Figura 37

Comprobación de la funcionalidad del sistema desarrollado para una red WSN.

```

jhhose@PORTEGE-Z30t-C: ~
12/04-02:13:06.603305  [**] [1:10000021:3] ATAQUE FLOOD_RS6 (RS) CON THCIpV6 [
**] [Classification: Potential Corporate Privacy Violation] [Priority: 1] {IPV
6-ICMP} fe80::75df:5d28:4739:10d9 -> ff02::1
12/04-02:13:06.603309  [**] [1:10000021:3] ATAQUE FLOOD_RS6 (RS) CON THCIpV6 [
**] [Classification: Potential Corporate Privacy Violation] [Priority: 1] {IPV
6-ICMP} fe80::75df:5d28:473a:10d9 -> ff02::1
12/04-02:13:06.603313  [**] [1:10000021:3] ATAQUE FLOOD_RS6 (RS) CON THCIpV6 [
**] [Classification: Potential Corporate Privacy Violation] [Priority: 1] {IPV
6-ICMP} fe80::75df:5d28:473b:10d9 -> ff02::1
12/04-02:13:06.603317  [**] [1:10000021:3] ATAQUE FLOOD_RS6 (RS) CON THCIpV6 [
**] [Classification: Potential Corporate Privacy Violation] [Priority: 1] {IPV
6-ICMP} fe80::75df:5d28:473c:10d9 -> ff02::1
12/04-02:13:06.603321  [**] [1:10000021:3] ATAQUE FLOOD_RS6 (RS) CON THCIpV6 [
**] [Classification: Potential Corporate Privacy Violation] [Priority: 1] {IPV
6-ICMP} fe80::75df:5d28:473d:10d9 -> ff02::1
12/04-02:13:06.603324  [**] [1:10000021:3] ATAQUE FLOOD_RS6 (RS) CON THCIpV6 [
**] [Classification: Potential Corporate Privacy Violation] [Priority: 1] {IPV
6-ICMP} fe80::75df:5d28:473e:10d9 -> ff02::1
12/04-02:13:06.603328  [**] [1:10000021:3] ATAQUE FLOOD_RS6 (RS) CON THCIpV6 [
**] [Classification: Potential Corporate Privacy Violation] [Priority: 1] {IPV
6-ICMP} fe80::75df:5d28:473f:10d9 -> ff02::1
12/04-02:13:06.603331  [**] [1:10000021:3] ATAQUE FLOOD_RS6 (RS) CON THCIpV6 [
**] [Classification: Potential Corporate Privacy Violation] [Priority: 1] {IPV
6-ICMP} fe80::75df:5d28:4740:10d9 -> ff02::1
12/04-02:13:06.603335  [**] [1:10000021:3] ATAQUE FLOOD_RS6 (RS) CON THCIpV6 [
**] [Classification: Potential Corporate Privacy Violation] [Priority: 1] {IPV
6-ICMP} fe80::75df:5d28:4741:10d9 -> ff02::1
12/04-02:13:06.603338  [**] [1:10000021:3] ATAQUE FLOOD_RS6 (RS) CON THCIpV6 [
**] [Classification: Potential Corporate Privacy Violation] [Priority: 1] {IPV
6-ICMP} fe80::75df:5d28:4742:10d9 -> ff02::1
12/04-02:13:06.603345  [**] [1:10000021:3] ATAQUE FLOOD_RS6 (RS) CON THCIpV6 [
**] [Classification: Potential Corporate Privacy Violation] [Priority: 1] {IPV
6-ICMP} fe80::75df:5d28:4743:10d9 -> ff02::1
12/04-02:13:06.629519  [**] [1:10000021:3] ATAQUE FLOOD_RS6 (RS) CON THCIpV6 [
**] [Classification: Potential Corporate Privacy Violation] [Priority: 1] {IPV
6-ICMP} fe80::75df:5d28:4744:10d9 -> ff02::1

jhhose@PORTEGE-Z30t-C: ~/Documentos/mininet/mininet-wifi/examples/WSN$ sudo atk6-f
lood_rs6 -s wlp2s0
Starting to flood with ICMPv6 router solicitation on wlp2s0 (Press Control-C to e
nd, a dot is printed for every 1000 packets):
.....^C
jhhose@PORTEGE-Z30t-C: ~/Documentos/mininet/mininet-wifi/examples/WSN$

jhhose@PORTEGE-Z30t-C: ~/Documentos/mininet/mininet-wifi/examples/WSN$
¡Alerta! Se encontraron anomalías en el tráfico de la red.
Esperando para la próxima iteración...
¡Alerta! Se encontraron anomalías en el tráfico de la red.
Esperando para la próxima iteración...
¡Alerta! Se encontraron anomalías en el tráfico de la red.
Esperando para la próxima iteración...
¡Alerta! Se encontraron anomalías en el tráfico de la red.
Esperando para la próxima iteración...
¡Alerta! Se encontraron anomalías en el tráfico de la red.
Esperando para la próxima iteración...
¡Alerta! Se encontraron anomalías en el tráfico de la red.
Esperando para la próxima iteración...

```

Nota. En la figura se muestra en la parte izquierda la salida de del IDS Snort, en la parte derecha superior se encuentra la ejecución del ataque y en la parte inferior se observa la salida del modelo Isolation Forest. Fuente: Autoría.

Ahora se inicia con un análisis de los resultados obtenidos durante la ejecución de las herramientas. A continuación, se especificarán algunas de las variaciones de rendimiento observadas durante la transferencia de datos los sensores de la red. Estos resultados ofrecen información que revela posibles patrones o tendencias dentro de la red, permitiendo que se evalúe la capacidad de la red frente a escenarios específicos de ataques.

En la Tabla 16 se observará la información acerca del tráfico que ocasionalmente presentan los sensores dentro de la red cuando es un envío de datos de normal.

Tabla 16

Tráfico normal de la red.

PRUEBAS DE TRÁFICO NORMAL	
Muestra 1	0.78 KB/s
Muestra 2	0.78 KB/s
Muestra 3	1.31 KB/s
Muestra 4	0.78 KB/s
Muestra 5	0.79 KB/s
Muestra 6	0.79 KB/s
Promedio de tráfico	0.87 KB/s

Nota. Según la información de la tabla se establece se considera un tráfico normal o carga moderada de los datos cuando este oscila ≈ 0.871 KB/s.

De acuerdo con las pruebas que se implementan ahora se observa cómo influye el primer ataque DoS **flood_rs6 -s**, recordando que es una inundación de paquetes ICMPv6 con solicitud de enrutador.

Tabla 17

Tráfico anómalo de la red.

PRUEBAS DE TRÁFICO ANÓMALO ATAQUE 1	
Muestra 1	9366.05 KB/s
Muestra 2	8970.36 KB/s
Muestra 3	9124.00 KB/s
Muestra 4	8580.74 KB/s
Muestra 5	9654.67 KB/s
Muestra 6	9371.60 KB/s
Promedio de tráfico	9177.90 KB/s

Nota. Se puede observar que existe diversas tasas de tráfico según las muestras, el promedio de tráfico anómalo será considerado ≈ 9177.90 KB/s.

Como se indicó anteriormente, la tabla anterior abordaba exclusivamente el primer ataque, y ahora se examina como impacta el segundo ataque DoS **flood_redir6 -F**, que implica una inundación de redirecciones ICMPv6.

Tabla 18

Tráfico anómalo de la red.

PRUEBAS DE TRÁFICO ANÓMALO ATAQUE 2	
Muestra 1	19283.10 KB/s
Muestra 2	19604.65 KB/s
Muestra 3	18975.13 KB/s
Muestra 4	17345.81 KB/s
Muestra 5	19391.72 KB/s
Muestra 6	18869.18 KB/s
Promedio de tráfico	18911.60 KB/s

Nota. Con esta información se considera un tráfico anómalo debido a que este fluctúa ≈ 18911.60 KB/s.

Por tanto, el enfoque proporciona una perspectiva más completa y precisa de como la red responde en las diferentes ocasiones a lo largo de la ejecución de las pruebas, aportando al entendimiento de resistencia y capacidad de recuperación frente a situaciones de ataques específicos.

Tabla 19

Resultados finales del tráfico de la red.

Pruebas			
Tráfico	Normal	Ataque 1	Ataque 2
Promedio	0.87 KB/s	9177.90 KB/s	18911.60 KB/s

Nota. De acuerdo con los datos obtenidos se deduce que el tráfico de la red WSN cuando se encuentra con una carga adecuado o idónea de datos se encuentra entre 0.871 KB/s, mientras que para el rango entre 9177.90 KB/s - 18911.60 KB/S es considerado como tráfico anómalo.

Mientras se encuentra en ejecución cada uno de los ataques que se explicaron anteriormente, se debe efectuar su detección, recordando que se usan las herramientas tanto del IDS como del modelo de inteligencia artificial.

Tabla 20

Cantidades estimadas para la detección de los ataques en la red.

DETECCIÓN DE ATAQUES		
	Snort	Isolation Forest
Falsos Positivos	9766	2261
Alertas Verdaderas	35671	43176
Total de Detecciones	45437	45437

Nota. Los valores detallados en la figura son un aproximado para el total de los paquetes que se podrían leer para la detección de anomalías.

Interpretando los resultados de la tabla previa, se pueden señalar los porcentajes en función de las pruebas de rendimiento del sistema.

Tabla 21

Resultados finales de la detección de los ataques en la red.

DETECCIÓN DE ATAQUES		
	Snort	Isolation Forest
Falsos Positivos	21.49%	4.97%
Alertas Verdaderas	78.51%	95.03%
Total de Detecciones	100%	100%

Nota. Los valores detallados en la figura son un aproximado para el total de los paquetes que se podrían leer para la detección de anomalías.

La tabla muestra el rendimiento de Snort e Isolation Forest para la detección de ataques. Los falsos positivos en Snort muestran una tasa de detección de 21.49% mientras que para Isolation Forest presenta una tasa muchos más baja con el 4.97%. Las alertas correctas del modelo de Isolation Forest presenta una tasa de 95.03% y Snort de 78.51%, sugiriendo de esta manera que es el más eficaz el modelo Isolation Forest.

Finalmente, la eficacia del sistema de detección tanto el IDS como el modelo de inteligencia artificial, va a depender de los diversos factores como el tipo de ataques, Snort es eficaz cuando se trata de reglas predefinidas, pero también produce falsos positivos si estas no se encuentran configuradas de manera correcta o el análisis del comportamiento o patrón de los ataques no es el esperado, Isolation Forest permite la adaptación a los patrones inusuales, por lo que suele ser más eficaz en la detección de las anomalías.

4.3. Discusión

En Diaz (2022) aplica el algoritmo de lógica difusa donde usa 27 reglas difusas basadas en conocimientos personales del laboratorio determinando así los parámetros para la calidad del agua en estado crítico, el sistema implementado tiene una precisión del 100% con un porcentaje de acierto de 94,4 según la muestra de la tabla de errores.

Por otro lado, en Salazar Cárdenas (2019) emplea la lógica difusa usado para tratar con la incertidumbre y la imprecisión de la toma de decisiones, controlando el momento de riego en los cultivos en funciones de los factores ambientales, permitiendo que el sistema de riego inteligente comparado con el riego manual reduce el consumo de agua.

Finalmente, Naula López (2021) utiliza el modelo de aprendizaje automático Random Forest el cual permite la clasificación del tráfico de la red, donde se determina las tramas normales y las maliciosas generadas por intrusos.

El sistema desarrollado en este proyecto usa un algoritmo basado en anomalías, con el cual se puede aplicar el modelo de Isolation Forest siendo un tanto similar a Random Forest, ya que este genera pequeños arboles de decisión mientras Isolation Forest genera particiones de manera aleatoria. Al aplicar este tipo de modelo es una ventaja debido a que está diseñado para que pueda detectar anomalías dentro del tráfico de red aún sin que este se encuentre entrenado y generando de esta manera las alertas para la prevención del ataque en el sistema.

Dentro del sistema diseñado en este proyecto se plantean las diversas pruebas con las que se comprueba el funcionamiento de este, según las tablas se establece el análisis detallado de acuerdo con los ataques que se implementan dentro de este, para el caso de tráfico normal este se encuentra en un rango de 0.87 KB/s, pero cuando se implementan los ataques su rango aumenta considerablemente en un rango de 9177.90 KB/s a 18911.60 KB/s, denominándolo en este caso como tráfico anómalo.

En base a los resultados del rendimiento tanto de Snort como de Isolation Forest proporciona una visión más clara sobre la eficacia de estas herramientas, en el análisis de la tasa de detección, Snort muestra un porcentaje de 21.49% para falsos positivos y para Isolation Forest un 4.97%, por otro lado, los verdaderos positivos presenta el 78.51% y 95.03% respectivamente, de este modo estableciendo que la herramienta más eficaz es el modelo Isolation Forest.

En conclusión, según los diversos trabajos que se han mencionado las soluciones al aplicar inteligencia artificial o modelos de aprendizaje automático son con respecto a los requisitos de

cada sistema, debido a que depende de las características y necesidades especificadas para el proyecto.

En este contexto, para el modelo que se utilizó se puede presentar características importantes, para implementar las soluciones para la detección de anomalías en un tráfico de red. En este ámbito el proyecto se desarrolló en base al modelo Isolation Forest, obteniendo resultados que validan su efectividad. Por ende, es esencial resaltar la importancia de establecer requisitos específicos para la introducción de inteligencia artificial o modelos de aprendizaje en proyectos. Estas herramientas pueden ser capaces de operar de manera efectiva cuando se definen los parámetros específicos, garantizando su integración exitosa en los proyectos propuestos.

CONCLUSIONES

- El análisis de vulnerabilidades y amenazas de las redes de sensores inalámbricos proporcionan una base bibliográfica fundamental para conocer sobre medidas de seguridad que se pueden aplicar para la protección de la red y representando también un inicio para la mitigación de riesgos.
- El establecer los requerimientos del sistema a desarrollar permite que se especifiquen los aspectos que influyen para la combinación de las herramientas a implementar ofreciendo así que operen de manera rápida y precisa ante los diversos escenarios que se pueden implementar dentro de las pruebas de funcionamiento.
- Los IDS (Sistema de Detección de Intrusos) cumple un rol importante dentro de la seguridad de las WSNs con respecto a los ataques DoS debido a que permiten que se detecte y alerte sobre comportamientos maliciosos dentro del tráfico de la red.
- El implementar el sistema en un ambiente simulado permite que las pruebas de funcionamiento sean un punto de partida para futuras implementaciones, donde se pueden considerar los posibles problemas antes de su desarrollo en un entorno operativo, para que de esta forma se asegure el funcionamiento correcto y óptimo.
- Según como se desarrolla la topología dentro del entorno de simulación, se sabe que toda la información de los nodos sensores será enviada hacia el nodo concentrador, nodo que permite que se mejore la capacidad de la detección de los ataques dentro de la red.
- La comprensión de las alertas que se pueden usar dentro de un IDS es un factor importante debido a que esto influye directamente sobre los registros de detección, información que es necesaria para poder aplicar el algoritmo basado en anomalías Isolation Forest.

- El uso del IDS dentro de la red desarrollada ayuda a que en un entorno real los administradores de red puedan detectar a tiempo ataques, llegando a aminorar el impacto que ocasionaría dentro de la red y manteniendo la disponibilidad de la misma.
- La implementación de la detección de ataques en una WSN puede tomar caminos diferentes, dado que se pueden basar en diversos ámbitos como el analizar patrones de comportamiento, monitoreo del tráfico y la detección de anomalías.
- Para obtener una detección eficaz de los ataques implementados dentro del proyecto, tanto el IDS como el modelo Isolation Forest deben de ser capaces de que procese y analice el tráfico de la red en tiempo real, pero se debe tener en cuenta el rendimiento del hardware debido a que se trabaja dentro un ambiente simulado, aplicando así un equilibrio entre los recursos disponibles.
- Gracias al uso del algoritmo basado en anomalías se puede mejorar la detección de los ataques DoS dentro de la WSN, dado que los modelos que pueden manejar un grupo de datos complejo para encontrar anomalías dentro del tráfico de la red como es el caso del modelo aplicado dentro del proyecto.
- El trabajar dentro de un entorno de simulación es posible que las diferentes configuraciones que se establecen sean creadas a partir de pruebas y errores, acciones que no podrían ser permitidas en un entorno real, dado que se debe tener en cuenta los diversos aspectos que pueden influir sobre la comunicación y transmisión de los nodos en conjunto.

RECOMENDACIONES

- Se recomienda que según el entorno en el que se trabaje para la simulación debe asegurarse que las configuraciones se den de manera correcta, debido a que es un punto clave para el comienzo de las futuras integraciones de todo el sistema a implementar.
- Para determinar que el modelo de inteligencia artificial a usar se debe investigar sobre los tipos de algoritmos que permiten la detección de ataques o anomalías dentro de la red, como se sabe algunos de ellos pueden ser algoritmos basados en reglas, en anomalías y en redes neuronales.
- Las reglas para la detección de ataques deben ser personalizadas para el tipo de ataque que se va a implementar dentro de la WSN.
- El IDS a implementar se debe adaptar a las condiciones de la red en la que se está trabajando, en el caso de este proyecto se configura para pueda trabajar en conjunto con Mininet-WiFi, procesando y analizando el tráfico de la red.
- El IDS y el modelo de Isolation Forest debe ser capaz de adecuarse para futuros cambios en el entorno de la red y a los nuevos posibles ataques DoS, es decir que deberían actualizarse y mejorarse de manera constante.
- Se recomienda se consideren las limitaciones de los recursos según el hardware que se use para poder implementar todo el sistema simulado, para así aplicar la optimización de los recursos y rendimiento del hardware usado.
- Para comprobar el funcionamiento correcto del IDS en el sistema se debe evaluar realizando pruebas en escenarios en los que, los ataques sean lo más realistas posibles y así saber la capacidad de detectar el ataque en la red.

BIBLIOGRAFÍA

- Ahmed, M. R., Aseeri, M., Kaiser, M. S., Zenia, N. Z., & Chowdhury, Z. I. (2015). A novel algorithm for malicious attack detection in UWSN. *2nd International Conference on Electrical Engineering and Information and Communication Technology, {iCEEiCT} 2015*.
<https://doi.org/10.1109/ICEEICT.2015.7307516>
- Ahmed, M. R., Huang, X., Sharma, D., & Cui, H. (2012). *Wireless Sensor Network: Characteristics and Architectures*. 6(12), 1398–1401.
<https://publications.waset.org/9345/wireless-sensor-network-characteristics-and-architectures>
- Anjum, F., Subhadrabandhu, D., & Sarkar, S. (2003). Signature based intrusion detection for wireless ad-hoc networks: a comparative study of various routing protocols. *2003 IEEE 58th Vehicular Technology Conference. VTC 2003-Fall (IEEE Cat. No.03CH37484)*, 58(3), 2152–2156 Vol.3. <https://doi.org/10.1109/VETEFCF.2003.1285405>
- Aronson, J. E. (2003). Expert Systems. *Encyclopedia of Information Systems*, 277–289.
<https://doi.org/10.1016/B0-12-227240-4/00067-8>
- Bace, R., & Mell, P. (2015). *NIST Special Publication on Intrusion Detection Systems Intrusion Detection Systems*.
- Benavides Arrieta, M. M. (2019). *Estudio de la Propagación de Ondas de Radio en la Banda Ism 2.4 Ghz en un Escenario Exterior Para el Diseño de Redes de Sensores Inalámbricas*.

https://repository.eafit.edu.co/bitstream/handle/10784/15375/Marledis_ArrietaBenavides_2019.pdf?isAllowed=y&sequence=2

Bertoldo, S., Paredes, M., Carosso, L., Allegretti, M., & Savi, P. (2019). *Empirical indoor propagation models for LoRa radio link in an office environment*. [https://www.complete-](https://www.complete-h2020network.eu/sites/default/files/upload/2019-)

[h2020network.eu/sites/default/files/upload/2019-11/08739575%20Empirical%20indoor%20propagation%20models%20for%20LoRa%20radio%20link%20in%20an%20office%20environment.pdf](https://www.complete-h2020network.eu/sites/default/files/upload/2019-11/08739575%20Empirical%20indoor%20propagation%20models%20for%20LoRa%20radio%20link%20in%20an%20office%20environment.pdf)

Burbano. (2017). Implementación de una red de sensores inalámbricos LPWAN mediante módulos LoRa para el monitoreo de la calidad del agua en 2 ríos. *Udistrital.Edu.Co*. <https://doi.org/http://hdl.handle.net/11349/6433>

Chowdhury, M., & Fazlul Kader, M. (2013). Security Issues in Wireless Sensor Networks: A Survey. *International Journal of Future Generation Communication and Networking*, 6(5), 97–116. <https://doi.org/10.14257/ijfgcn.2013.6.5.10>

Dalal, B., & Kukarni, S. (2021). *Wireless Sensor Networks: Applications*. IntechOpen. <https://doi.org/10.5772/intechopen.97079>

Díaz, J. W. (2022). *Sistema de monitoreo inteligente basado en tecnología IOT e inteligencia artificial aplicado a la crianza de alevines, en la parroquia el Playón de San Francisco perteneciente al cantón Sucumbíos provincia Sucumbíos*. <http://repositorio.utn.edu.ec/handle/123456789/12950>

- Effrosynidis, D. (2020). *Outlier Detection — Theory, Visualizations, and Code* / by Dimitris Effrosynidis / *Towards Data Science*. <https://towardsdatascience.com/outlier-detection-theory-visualizations-and-code-a4fd39de540c>
- Faircloth, J. (2014). Networks. *Enterprise Applications Administration*, 27–79. <https://doi.org/10.1016/B978-0-12-407773-7.00002-8>
- Fei Tony Liu, Kai Ming Ting, & Zhi-Hua Zhou. (2008). *Isolation Forest*.
- Felipe, E., & Angel, A. (2020). *Desarrollo de un sistema de adquisición, transmisión y monitoreo para una red de sensores de precipitación*. https://ciencia.lasalle.edu.co/ing_automatizacion/275/
- Galo Andy, Gusñay, J. L., Ovaco, M. J., Puma, A. F., & Uvidia, A. S. (2023). *Análisis comparativo de modelos de pérdida de trayectoria de propagación para comunicación móvil en Riobamba*. <https://dialnet.unirioja.es/descarga/articulo/8822792.pdf>
- Gustavsson, V. (2019). *Machine Learning for a Network-based Intrusion Detection System : An application using Zeek and the CICIDS2017 dataset*.
- Gutiérrez Portela, F., Almenares Mendoza, F., Calderón Benavides, L., & Romero Riaño, E. (2021). *Security perspective of wireless sensor networks*. 20(3), 189–202. <https://doi.org/10.18273/revuin.v20n3-2021014>
- Haro Bermejo Francisco. (2015). *Detección de intrusiones con Snort*. <https://docplayer.es/7991177-Universitat-oberta-de-catalunya-postgrado-seguridad-en-redes-y-sistemas-proyecto-fin-de-postgrado-deteccion-de-intrusiones-con-snort.html>

- Heaton, J. (2016). Comparing dataset characteristics that favor the Apriori, Eclat or FP-Growth frequent itemset mining algorithms. *Conference Proceedings - IEEE SOUTHEASTCON, 2016-July*. <https://doi.org/10.1109/SECON.2016.7506659>
- Heredia Rivadeneria, E. A., & Lucero Andrade, F. P. (2021). *Diseño e implementación de una red inalámbrica de sensores con tecnología {LoRa} para monitoreo industrial orientado a {OPC} de arquitectura unificada Autores: Cuenca -Ecuador 10 de marzo de 2021*. [https://dspace.ucuenca.edu.ec/bitstream/123456789/35875/1/Trabajo de Titulación.pdf](https://dspace.ucuenca.edu.ec/bitstream/123456789/35875/1/Trabajo%20de%20Titulaci3n.pdf)
- HN Lakshmi, Anand Santesh, & Sinha Somnath. (2019). *Flooding Attack in Wireless Sensor Network-Analysis and Prevention*. https://www.researchgate.net/publication/334389404_Flooding_Attack_in_Wireless_Sensor_Network-Analysis_and_Prevention
- ISO/IEC. (2013). *ISO/IEC 27001:2013*. <https://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/05/45/54534.htm>
- 1
- ISO/IEC/ IEEE 29148. (2018). Systems and software engineering — Life cycle processes — Requirements engineering Ingénierie. *Iso/Iec/Ieee 29148:2018, 2012(40)*.
- Janezko, B., & Srivastava, G. (2022). The use of deep learning in image analysis for the study of oncology. *Internet of Multimedia Things (IoMT): Techniques and Applications*, 133–150. <https://doi.org/10.1016/B978-0-32-385845-8.00011-3>
- Jyothsna V, Rama Prasad V.V, & Munivara Prasad K. (2011). *A Review of Anomaly based Intrusion Detection Systems*.

- Kanimozhi, V., & Jacob, T. P. (2019). Artificial Intelligence based Network Intrusion Detection with Hyper-Parameter Optimization Tuning on the Realistic Cyber Dataset {CSE}-{CIC}-{IDS}2018 using Cloud Computing. *2019 International Conference on Communication and Signal Processing (ICCSP)*, 33–36. <https://doi.org/10.1109/ICCSP.2019.8698029>
- Kaur, R., & Kaur Sandhu, J. (2021). A Study on Security Attacks in Wireless Sensor Network. *2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, 850–855. <https://doi.org/10.1109/ICACITE51222.2021.9404619>
- Keerthika, M., & Shanmugapriya, D. (2021). *Wireless Sensor Networks: Active and Passive attacks - Vulnerabilities and Countermeasures*. 2(2), 362–367. <https://doi.org/10.1016/j.gltip.2021.08.045>
- Khaled AlSedrah Miriam. (2017). *Artificial Intelligence*. https://www.researchgate.net/publication/323498156_Artificial_Intelligence
- Khaled, Z. El, Ajib, W., & Mcheick, H. (2022). *Log Distance Path Loss Model: Application and Improvement for Sub 5 GHz Rural Fixed Wireless Networks*. 10, 52020–52029. <https://doi.org/10.1109/access.2022.3166895>
- Khan, W. Z., Aalsalem, M. Y., Saad, M. N. B. M., & Xiang, Y. (2013). Detection and Mitigation of Node Replication Attacks in Wireless Sensor Networks: A Survey. *Http://Dx.Doi.Org/10.1155/2013/149023, 2013*. <https://doi.org/10.1155/2013/149023>
- Kizza, J. M. (2017). *Introduction to Computer Network Vulnerabilities*. 87–103. https://doi.org/10.1007/978-3-319-55606-2_4

- Kotu, V., & Deshpande, B. (2019). Deep Learning. *Data Science*, 307–342.
<https://doi.org/10.1016/B978-0-12-814761-0.00010-1>
- Kuan, F. (2020). *LoRa - Un estándar de radio abierto de LPWAN*.
<https://www.mokosmart.com/es/lora/#:~:text=La%20distancia%20entre%20el%20transmis,or%20y%20el%20receptor,rurales%29%20dependiendo%20del%20entorno%20y%20las%20zonas%20urbanizadas.>
- Kumar, Y., Kumar, K., & Munjal, R. (2011). Wireless Sensor Networks and Security Challenges
 Rajiv Munjal. *International Journal of Computer Applications*®.
- Kunal, & Dua, M. (2019). Machine Learning Approach to IDS: A Comprehensive Review. *2019 3rd International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, 117–121. <https://doi.org/10.1109/ICECA.2019.8822120>
- Kurniawan, M. T., & Yazid, S. (2020). Mitigation and Detection Strategy of DoS Attack on Wireless Sensor Network Using Blocking Approach and Intrusion Detection System. *2nd International Conference on Electrical, Communication and Computer Engineering, ICECCE 2020*. <https://doi.org/10.1109/ICECCE49384.2020.9179255>
- Laybats, C., & Tredinnick, L. (2016). *Information security*. 33(2), 76–80.
<https://doi.org/10.1177/0266382116653061>
- Leal Ávila Luis Daniel. (2019). *Topología de árbol* . <https://idoc.pub/documents/idocpub-6klzeqxq0elg>

Locicero Giorgio. (2020). *Suricata*.

https://www.researchgate.net/publication/344292913_Suricata_review_and_attack_sceneries#pf3

López Ávila, L., Acosta Mendoza, N., & Gago, A. (2019). *Detección de anomalías basada en aprendizaje profundo: Revisión*.

MAGERIT. (2012). *Metodología de Análisis y Gestión de Riesgos de los Sistemas de Información*.

<http://administracionelectronica.gob.es/>

Microchip. (2021). *RN2903*.

<https://ww1.microchip.com/downloads/aemDocuments/documents/WSG/ProductDocuments/DataSheets/RN2903-Low-Power-Long-Range-LoRa-Technology-Transceiver-Module-DS50002390K.pdf>

Mohammed Riyadh, A. (2016). *Data Confidentiality in the Internet of Things*.

<https://doi.org/10.13140/RG.2.2.19150.87366>

Mohammed Ziyad. (2019). *Artificial Intelligence Definition, Ethics and Standards*.

https://www.researchgate.net/publication/332548325_Artificial_Intelligence_Definition_Ethics_and_Standards#pf4

Mollet, M. S., & Kisangiri, M. (2014). *Comparison of Empirical Propagation Path Loss Models for Mobile Communication*. <https://core.ac.uk/download/pdf/234644867.pdf>

Muñoz Barragán Sócrates Nelson, Toquica Álvarez Andrés Felipe, & Padilla José Bestier. (2017). *Red de sensores inalámbricos para el monitoreo de variables microclimáticas en el Relicto Vegetal Cedro Rosado.*

Naula López, E. R. (2021). *Diseño e Implementación de un Sistema de Detección de Intrusiones para redes Wifi usando herramientas de Big Data y Machine Learning.*
<https://repositorio.espe.edu.ec/bitstream/21000/25221/1/T-ESPE-044601.pdf>

Nweke, L. (2017). *Using the CIA and AAA Models to explain Cybersecurity Activities.*

Obaid, H. S., & Abeed, E. H. (2020). *DoS and DDoS Attacks at OSI Layers.* 2(8).
<https://doi.org/10.5281/zenodo.3610833>

Obaidat, M. S., & Misra, S. (2014). Security issues in wireless sensor networks. *Principles of Wireless Sensor Networks*, 222–247. <https://doi.org/10.1017/CBO9781139030960.011>

Ojeda Guerra Carmen Nieves. (2020). *Implementación de una plataforma redundante de control SDN-IoT.*

O'Mahony, G. D., Curran, J. T., Harris, P. J., & Murphy, C. C. (2020). *Interference and Intrusion in Wireless Sensor Networks.* 35(2), 4–16. <https://doi.org/10.1109/MAES.2020.2970262>

Panzuela Perez, F. (2022). *Estudio y validación de Mininet-WiFi en base a pruebas reales.*
<https://riunet.upv.es/bitstream/handle/10251/187240/Panzuela%20-%20Estudio%20y%20validacion%20de%20Mininet-WiFi%20en%20base%20a%20pruebas%20reales.pdf?isAllowed=y&sequence=1>

- Romero Castro, M. I., Figueroa Morán, G. L., Vera Navarrete, D. S., Álava Cruzatty, J. E., Parrales Anzúles, G. R., Álava Mero, C. J., Murillo Quimiz, Á. L., & Castillo Merino, M. A. (2018). *Introducción a la Seguridad Informática y el Análisis de Vulnerabilidades*. https://books.google.com.ec/books?id=5Z9yDwAAQBAJ&printsec=frontcover&dq=que+e+s+el+modelo+cia+en+telecomunicaciones&hl=es&sa=X&ved=2ahUKEwjFw_3J4Ib6AhULgIQIHbaaAGYQ6AF6BAgFEAI#v=onepage&q&f=false
- Sadaf, S., Iqbal, S., Saba, A., & KamarMohsin, Md. (2017). An extended adaptive process model for agile software development methodology. *2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT)*, 1373–1378. <https://doi.org/10.1109/ICICICT1.2017.8342770>
- Sahu, S. S., Priyadarshini, P., & Bilgaiyan, S. (2014). Curbing distributed denial of service attack by traffic filtering in wireless sensor network. *5th International Conference on Computing Communication and Networking Technologies, ICCCNT 2014*. <https://doi.org/10.1109/ICCCNT.2014.6963043>
- Salazar Cárdenas, L. J. (2019). *Diseño de un Sistema de Riego Inteligente para Cultivos de Hortalizas basado en Fuzzy Logic en la Granja la Pradera de la Universidad Técnica del Norte*. <http://repositorio.utn.edu.ec/handle/123456789/9137>
- Sen, J. (2010). *Routing Security Issues in Wireless Sensor Networks: Attacks and Defenses*.
- Shafiei, H., Khonsari, A., Derakhshi, H., & Mousavi, P. (2014). Detection and mitigation of sinkhole attacks in wireless sensor networks. *Journal of Computer and System Sciences*, 80(3), 644–653. <https://doi.org/10.1016/J.JCSS.2013.06.016>

Silnik, A., Dugarte, N., Pérez, S., & Facchini, H. (2021). *Redes de Sensores Inalámbricos (WSN)*.

<http://academiasutnmza.com/wp-content/uploads/2021/08/Curso-de-Redes-de-Sensores-Inalambricos-WSN.pdf>

Sommer, R. (2003). *Bro: An Open Source Network Intrusion Detection System*.

Suganya, E., Sountharajan, S., Shandilya, S. K., & Karthiga, M. (2019). Chapter 5 - IoT in Agriculture Investigation on Plant Diseases and Nutrient Level Using Image Analysis Techniques. In V. E. Balas, L. H. Son, S. Jha, M. Khari, & R. Kumar (Eds.), *Internet of Things in Biomedical Engineering* (pp. 117–130). Academic Press. <https://doi.org/10.1016/B978-0-12-817356-5.00007-3>

Sun, D., Wang, W., Lu, J., & Lin, Z. (2010). Design of WSN nodes and network performance analysis in a tea plantation. *{IET} International Conference on Wireless Sensor Network 2010 ({IET}-{WSN} 2010)*, 144–147. <https://doi.org/10.1049/cp.2010.1043>

Teixeira, D., Assunção, L., Pereira, T., Malta, S., & Pinto, P. (2019). OSSEC IDS extension to improve log analysis and override false positive or negative detections. *Journal of Sensor and Actuator Networks*, 8(3). <https://doi.org/10.3390/JSAN8030046>

Tiwari Mahit, Kumar Raj, Bharti Akash, & Kishan Jai. (2017). *INTRUSION DETECTION SYSTEM*.

https://www.researchgate.net/publication/316599266_INTRUSION_DETECTION_SYSTE
M

Torres, J. M., Pinto-Mangones, Á., Macea A, M. R., Pérez-García, N. A., & Marian Rujano, L. (2016). *PATH LOSS PREDICTIONMODEL FOR WLAN OPERATING AT 2.4 GHZ AND 5.8*

GHZ, IN INDOOR ENVIRONMENTS OF COMMERCIAL BUILDINGS.

<http://ve.scielo.org/pdf/uct/v20n78/art04.pdf>

Trevino Cortes, J. T. (2003). *Propagacion de RF en las bandas: LF, MF, HF, VHF, UHF y VHF.*

http://catarina.udlap.mx/u_dl_a/tales/documentos/lem/trevino_c_jt/

Vallero, A., Savino, A., Chatzidimitriou, A., Kaliorakis, M., Kooli, M., Riera, M., Anglada, M.,

Di Natale, G., Bosio, A., Canal, R., Gonzalez, A., Gizopoulos, D., Mariani, R., & Di Carlo,

S. (2019). SyRA: Early system reliability analysis for cross-layer soft errors resilience in memory arrays of microprocessor systems. *IEEE Transactions on Computers*, 68(5), 765–

783. <https://doi.org/10.1109/TC.2018.2887225>

Vasudeva, A., & Sood, M. (2022). *On the vulnerability of the mobile ad hoc network to transmission power controlled Sybil attack: Adopting the mobility-based clustering.*

<https://doi.org/10.1016/j.jksuci.2022.04.020>

Vázquez Rodas, A., Astudillo Salinas, F., & Minchala, L. I. (2021). *Aplicación de tecnologías inalámbricas al monitoreo climatológico en la cuenca del Río Paute.* 9(17), 89–96.

<https://www.riti.es/ojs2018/inicio/index.php/riti/article/view/272>

Vlajic, N., Stevanovic, D., & Spanogiannopoulos, G. (2011). *Strategies for improving performance of IEEE 802.15.4/ZigBee WSNs with path-constrained mobile sink(s).* 34(6),

743–757. <https://doi.org/10.1016/j.comcom.2010.09.012>

Woolf, B. P. (2009). Machine Learning. *Building Intelligent Interactive Tutors*, 221–297.

<https://doi.org/10.1016/B978-0-12-373594-2.00007-1>

- Yan, W. Q. (2021). *Computational Methods for Deep Learning*. <https://doi.org/10.1007/978-3-030-61081-4>
- Yong-Min, L., Shu-Ci, W., & Xiao-Hong, N. (2009). The Architecture and Characteristics of Wireless Sensor Network. *2009 International Conference on Computer Technology and Development, 1*, 561–565. <https://doi.org/10.1109/ICCTD.2009.44>
- Yordanov, V. (2021). *Things to Consider When Picking A LoRaWAN® Gateway*. RAKwireless News Hub. <https://news.rakwireless.com/things-to-consider-when-picking-a-lorawan-gateway/>
- Zulfadhilah, M. (2017). *The Importance Of Securing Digital Data*. 431–435. <https://doi.org/10.2991/smichs-17.2017.53>

ANEXOS

Anexo A

Objetivo: Analizar criterios establecidos para el diseño del sistema de detección de intrusos a implementarse en una red WSN con la aplicación de la inteligencia artificial, recopilando información de referencias bibliográficas, para definir la arquitectura de la red deseada para el proyecto.

Tabla 22

Análisis de referencias bibliográficas para la definición de la arquitectura del proyecto.

Softwares	Referencias bibliográficas	Simulación de redes WSN	Soporte de protocolos	Interoperabilidad de software	Eficiencia de procesamiento	Sistema Operativo Compatible	Adaptación de configuraciones
OMNeT++	Muñoz Choez, V. H. (2018). <i>Análisis del modelado de redes de sensores inalámbricos mediante validación de la plataforma de simulación OMNeT</i> . (2021). <i>OMNeT++ Installation Guide</i> .	Dentro De OMNeT++ se encuentra un Castalia que es un simulador robusto la cual permite la simulación de varios tipos de redes.	Soporta todos los protocolos de red estándar.	OMNeT++ puede presentar su integración con otras aplicaciones como MATLAB, Simulink, NS-3, entre otros.	Este aspecto va a depender de parámetros como el tamaño de la simulación y la complejidad de esta.	Esta herramienta está disponible para distribuciones como Windows, Ubuntu, MacOS,	Dentro del programa se encuentran archivos que se pueden conservar su configuración o comenzar con su edición.

	https://doc.omnetpp.org/omnetpp/InstallGuide.pdf					Fedora, entre otros.	
Mininet	Lantz, B., Heller, B., & Mckeown, N. (n.d.). <i>A Network in a Laptop: Rapid Prototyping for Software-Defined Networks.</i>	Permite la creación de redes SDN.	Soporta de protocolos de red estándar	Mininet presenta una flexibilidad y compatibilidad con una gama grande de herramientas, softwares de red. Por ejemplo, combinaciones con Docker o QEMU.	Se considera muy eficiente con los recursos de la herramienta, pero se deben considerar las limitaciones y requisitos del entorno para simular.	Su instalación se puede realizar en un sistema de virtualización como por ejemplo para VirtualBox herramienta que puede ser instalado en varios S.O entre ellos se encuentra MacOS, Windows y Linux.	Dentro de la herramienta se pueden conseguir varios scripts donde se puede realizar diferentes cambios que se adapten de acuerdo a las necesidades que requiere la simulación a desarrollar.
NS-3	NS-3. (2023). <i>NS-3.</i> https://www.nsnam.org/	Este programa se usa para simular redes,	Soporta todos los protocolos	NS-3 es un software independiente	Este programa diseñado para realizar	La herramienta es compatible	Es posible realizar los ajustes necesarios para personalizar los

además de que existen características y herramientas que funcionan en este tipo de redes WSN. de estándar. red , pero se puede realizar una integración con otros programas para poder extender las funcionalidad es de este, además de que estos se deben considerar de acuerdo con el proyecto a desarrolla. simulaciones precisas de redes de comunicación requiere un mayor uso de recursos para procesar funciones comparando un programa de virtualización simple. con algunos aspectos de la simulación para poder adaptarlo al entorno que se requiere. de los sistemas operativos como Linux, MacOS y Windows.

NS-2 *The Network Simulator - ns-2.* (2023). Retrieved May 27, 2023, from <https://www.isi.edu/nsnam/ns/> Aboelela, Emad., & Peterson, L. L. (2011). *Network simulation experiments manual.* NS-2 puede realizar simulaciones con propósitos generales, sin embargo, se establece que si se puede dar la virtualización de una red WSN. Dentro de esta plataforma se puede aplicar los protocolos de enrutamiento y protocolos de acceso al medio en redes inalámbricas. Este simulador proporciona interoperabilidad con diversos softwares y herramientas, como por ejemplo usar herramientas externas como Python, La eficiencia de procesamiento dependerá de los factores que interfieran dentro de la simulación, la cantidad de nodos, enlaces y protocolos implementados. Esta plataforma originalmente se desarrolló para sistemas operativos basados en Unix, también está disponible para MacOS, y para Windows. La adaptabilidad de NS-2 es muy grande, dado que es configurable permitiendo que se adapte la simulación de diversos escenarios y personalizarlo de acuerdo con las necesidades.

				MATLAB, OMNeT++, QualNet y OPNET, ampliando las capacidades de análisis.		puede usarse con emuladores o máquinas virtuales.	
NetSim	NetSim. (2023). <i>NetSim User Manual Contents.</i>	La plataforma puede desarrollar redes, entre ellas la red de sensores inalámbricos, ya que proporciona un amplio entorno para modelar y analizar el comportamiento de estas redes en diversos ambientes.	Soporta todos los protocolos de red estándar.	La interoperabilidad de este programa es por medio de API e interfaces estándar, además de considerar que se puede importar y exportar las configuración es de red.	La eficiencia de procesamiento es de acuerdo con la optimización de algoritmos y modelos, donde se configure el tiempo de simulación para la precisión y eficiencia.	Se encuentra disponible para las distribuciones de Windows 10,8 7 además de usar los programas de virtualización como VirtualBox o VMware, para luego instalar una distribución de Windows.	La herramienta proporciona la configuración de varios de sus aspectos como por ejemplo protocolos, escenarios personalizados, además de sus enlaces y canales.
Cooja	Contiki-NG. (2023). <i>Cooja — Contiki-NG documentation.</i> https://docs.contiki-	Cooja es parte de un entorno más grande como lo es	Soporta protocolos WSN	Este programa permite el uso de otros softwares o	La eficiencia de procesamiento se basa en los	Algunas de las distribuciones en las que se puede ejecutar	Se pueden hacer configuraciones para personalizar los parámetros necesarios,

ng.org/en/develop/do
c/platforms/cooja.ht
ml

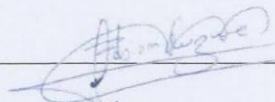
Contiki el cual
se orienta por
aplicaciones de
IoT y WSN.

herramientas
que pueden
operar en
conjunto
como, por
ejemplo, los
entornos de
desarrollo
integrados,
OMNeT++ y
también que
existe la
posibilidad de
realizar
importaciones
con
herramientas
como
MATLAB o
Python.

parámetros de
esta
plataforma es
en Linux,
MacOS y
Windows en
este último se
debe tener en
cuenta que se
debería de
utilizar los
entornos de
virtualización
entre ellos
puede ser
VirtualBox o
Docker.

como el protocolo y
algoritmos que se
pueden utilizar en la
simulación de WSN.

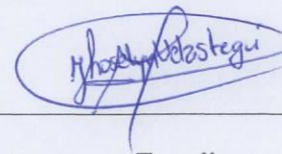
Revisor por:



Director

Msc. Fabián Geovanny Cuzme Rodríguez

Elaborado por:



Estudiante

Jhoselyn Lizeth Velastegui Morales

Anexo B

Objetivo: Establecer las características a considerar del ataque a implementar en una WSN aspecto requerido para que se desarrolle el IDS, permitiendo que este ayude con las evaluaciones necesarias para obtener resultados óptimos en su funcionamiento.

Tabla 23

Análisis de referencias bibliográficas para la definición del ataque a implementar.

Ataque	Referencia bibliográfica	Capa	Despliegue del ataque	Detección del ataque	Mitigación/Prevención del ataque
Denegación de servicio distribuido (DDoS)	Upadhyay, R., Khan, S., Tripathi, H., & Bhatt, U. R. (2016). Detection and prevention of DDOS attack in WSN for AODV and DSR using battery drain. 2015 <i>International Conference on Computing and Network Communications</i> ,	Las capas en las que puede afectar son las de enlace de datos, red y transporte, pero en todas se considera que puede variar la forma de operar y afectar en ella cumpliendo con su objetivo.	Técnica Flooding	Aplica el método BatteryGetRemaininngCharge el cual recupera la energía restante de la batería de cada nodo en la cual se tiene definidos los niveles tanto bajos como altos en los cuales se manejan	Dada la detección de los nodos maliciosos, estos se incluirán dentro de una lista negra, además de que se proporcionó un determinado tiempo de vida de la red, con esto al proceso de apagado los nodos maliciosos no consumirán más batería de otros nodos de la red.

CoCoNet 2015, 446–451.

<https://doi.org/10.1109/COCONET.2015.7411224>

CISA. (2020). *DDoS Quick Guide*.

<https://www.cisa.gov/sites/default/files/publications/DDoS%20Quick%20Guide.pdf>

los nodos en el caso de que se observe una variación dentro de esto se establece que ese nodo puede ser víctima del ataque.

Inundación

Dharini, N., Balakrishnan, R., & Renold, A. P. (2015). Distributed detection of flooding and gray hole attacks in Wireless Sensor Network. *2015 International Conference on Smart Technologies and* Este tipo de ataque es común que aplique dentro de la capa de red cumpliendo el objetivo de que se agoten los recursos, pero también se puede realizarlo a nivel de capa dos de

Agotar los recursos de la red hablando en aspectos de ancho de banda y energía.

Se usa una predicción de energía aplicando un algoritmo basado en el aprendizaje.

La forma en que se realiza un control es aplicando la lectura de la energía de los nodos, permitiendo que se ejecute una comparación entre esos nodos para detectar las anomalías.

Management for enlace de datos
Computing, crenado
Communication, congestiones en los
Controls, Energy and enlaces para afectar
Materials, ICSTM 2015 - de esta manera la
Proceedings, 178–184. comunicación entre
<https://doi.org/10.1109/ICSTM.2015.7225410> nodos.
 Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., & Cayirci, E. (2002). A survey on sensor networks. *IEEE Communications Magazine*, 40(8), 102–105.
<https://doi.org/10.1109/MCOM.2002.1024422>

Falsificación de identidad

Kalabarige, L., & Maringanti, H. B. (2021). A Survey on Este tipo de ataques se puede hacer con el cambio de Según los recursos de la red, algunas de ellas son limitadas Para este tipo de ataque se lo puede tratar dentro de Considerando el uso de intercambio de claves basado en la identidad se puede

Identity-Based Security direcciones MAC, hablando de varios ámbitos compartir claves secretas, in *Wireless Sensor afectando así a la comunicación, como los protocolos claves por pares, clave de Networks. Lecture Notes capa de enlace, almacenamiento y de autenticación o el clúster, claves de grupo y in Networks and permitiendo cálculo de nodos, uso de métodos claves de sesión. Systems, 134, 487–498. interceptar paquetes por lo que es fácil criptográficos* https://doi.org/10.1007/978-981-15-5397-4_50/COVER o enviar datos falsos que el atacante donde se use el intercambio de igual manera se transmisiones entre claves secretas para Faria, D. B., & Cheriton, considera que se los nodos. los nodos D. R. (2006). Detecting puede aplicar en la asegurando la identity-based attacks in capa de red donde se autenticidad de los wireless networks using manipule la nodos. signalprints. *WiSE 2006 dirección IP o la - Proceedings of the 5th identificación del ACM Workshop on nodo. Wireless Security, 2006, 43–52.* <https://doi.org/10.1145/1161289.1161298>

Retransmisión

Tanabe, N., Kohno, E., El ataque de En las WSN los En este tipo de En el artículo menciona que & Kakuda, Y. (2013). A retransmisión se ataques pueden ataque también se se puede usar el filtro Bloom

path authenticating puede realizar sustraer de manera puede realizar la el cual permite que los datos
 method using bloom dentro de la capa de fácil los paquetes implementación de se reduzcan en tamaño para
 filters against enlace de datos retransmitidos por autenticación luego reconstruirlos y
 impersonation attacks on interceptando los un nodo permitiendo la hacerlos más manejables por
 relaying nodes for paquetes retransmisor, verificación de los lo que actuaría como una
 wireless sensor transmitidos, usando un nodo datos o información función de hash que viene
 networks. *Proceedings -* información que malicioso también que se envíen a hacer como un cifrado con
International puede ser enviada a puede manipular la través de ellos. que se requerirá de
Conference on otro lugar de la red, información como autenticación por parte de
Distributed Computing considerando que se las direcciones IP, cada nodo.
Systems, 357–361. falsifican varias además de su ID de
<https://doi.org/10.1109/ICDCSW.2013.34> cosas como las origen y destino.
 direcciones MAC,
 permitiendo
 engañar a otros
 nodos, por otro lado,
 la capa de red
 también se ve
 afectada ya que se
 puede manipular
 aspectos del
 encabezado de los

paquetes para
falsificar la
dirección IP o
modificar la
secuencia de los
paquetes.

Revisor por:

Elaborado por:



Director
Msc. Fabián Geovanny Cuzme Rodríguez

Estudiante
Jhoselyn Lizeth Velastegui Morales

Anexo C

Objetivo: Establecer las características a considerar del algoritmo de detección a implementar en una WSN aspecto requerido para que se aplique la inteligencia artificial en conjunto con el IDS.

Tabla 24

Análisis de referencias bibliográficas para la definición del algoritmo de detección del proyecto.

Modelos	Referencias bibliográficas	Descripción	Simplicidad de implementación	Procesamiento	Precisión	Capacidad	Interoperabilidad	Sensibilidad de datos
Apriori	Heaton, J. (2016). Comparing dataset characteristics that favor the Apriori, Eclat or FP-Growth frequent itemset	Este algoritmo usa la minería de datos en la que se pueden encontrar patrones de asociación dentro de un grupo de datos, su	Al usar un proceso iterativo requiere de un gran consumo computacional según el conjunto de datos	La velocidad del procesamiento en un grupo pequeño de datos es alta, a diferencia de que si es un grupo grande	Se considera una precisión baja en detección de anomalías, porque no se encuentra diseñado con ese objetivo.	No tiene una buena capacidad para los datos debido a que se vuelven con menos eficiencia con una gran	Se puede realizar su implementación con diversos lenguajes de programación como por ejemplo Python o	Permite que se pueda identificar las reglas o patrones en mínimas cantidades o que no sean obvios.

<p>mining algorithms. Conference Proceedings - IEEE SOUTHEAST CON, 2016- July. https://doi.org/10.1109/SEC ON.2016.7506659</p>	<p>objetivo principal es observar con qué frecuencia se llegan a producir ciertos conjuntos de elementos de acuerdo a las bases de datos.</p>	<p>este se volverá lenta.</p>	<p>cantidad de herramientas características de visualización.</p>		
<p>FP-Growth</p>	<p>Frequent Pattern Growth este algoritmo también utiliza minería de datos al igual que A priori, Se considera más eficiente que Apriori, pero su implementación tiene complicación cuando se</p>	<p>Se considera mucho más rápido que Apriori.</p>	<p>Este tipo de modelo no es adecuado para la detección de anomalías.</p>	<p>Cuando se tratan de datos con complejidad de procesamiento se considera con eficiencia baja.</p>	<p>Se pueden usar bibliotecas para implementar el código entre ellas se encuentra scikit-learn o MLib. Permite que se identifiquen grupos de elementos que se muestran con frecuencia o incluso si no se consideran comunes dentro del</p>

patrones de lo hace asociación en desde cero. un conjunto de datos, este se diferencia con Apriori porque usa una estructura de datos de árbol permitiendo que se acelere el proceso de búsqueda para los patrones frecuentes.

grupo de datos.

Isolation Forest	Effrosynidis, D. (2020). <i>Outlier Detection — Theory, Visualizations</i>	Este algoritmo se basa en la detección de anomalías basado en arboles donde	Se considera que su implementación es sencilla	Su velocidad de procesamiento es alta incluso en grandes	Es aplicado para la detección de anomalías y es efectiva para	No le afecta la dimensión de los datos, a comparación con otros modelos.	Se usa en el lenguaje de programación Python, además de la posibilidad	Su rendimiento se califica como alto incluso los
------------------	--	---	--	--	---	--	--	--

<p>One-Class SVM</p>	<p>, and Code / by Dimitris Effrosynidis / Towards Data Science. https://towardsdatascience.com/outlier-detection-theory-visualizations-and-code-a4fd39de540c</p>	<p>su objetivo es aislar anomalías de un conjunto de datos, usa la división de las características con árboles de decisión. Algoritmo de Su aprendizaje automático para la detección de anomalías, este se entrena con un conjunto de datos normales y luego usarlo</p>	<p>conjuntos de los datos atípicos. Su velocidad de procesamiento depende con el tamaño del grupo de datos a analizar. Alta precisión de detección de anomalías</p>	<p>Este modelo puede permitirse el manejo de la alta dimensión de los datos.</p>	<p>del uso de datos no son bibliotecas lineales. como sklearn o scikit-learn. Se considera baja en la aplicación de modelos complejos. La sensibilidad de los datos es baja debido a que funciona mejor con los datos lineales.</p>
	<p>datos. atípicos.</p>	<p>datos. atípicos.</p>	<p>del uso de datos no son bibliotecas lineales. como sklearn o scikit-learn.</p>		

para la
identificación
de las
observaciones
inusuales.
Su
complejidad
es media
debido a que
se requiere
conocimiento
sobre
aprendizaje
automático.

DBSCAN

Density- Su Su velocidad Aplicado para
Based Spatial implementa de la detección
Clustering of ción es procesamiento de conjuntos
Applications complicada depende del de anomalías,
with Noise dado que tamaño del pero podría
algoritmo de pueden grupo de datos ser menos
clustering realizarse y la preciso para

Su capacidad
es baja debido
a la
dimensionalidad
de los
datos.

Puede usar Realiza
varios pruebas y
lenguajes de experimentos
programación para la
o librerías de determinación
aprendizaje de valores
automático. óptimos para

usado para la detección de anomalías, donde los puntos que se encuentran aislados se consideran como anomalía. Su implementación es complicada dado que pueden realizarse cambios de parámetros y compresión de densidad de los datos.

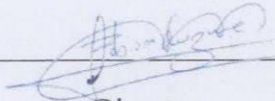
el grupo de datos.

CNN	<p>Li, W. Q. (2021). <i>Computational Methods for Deep Learning</i>. https://doi.org/10.1007/978-3-030-61081-4</p>	<p>Convolutional Neural Network son un tipo de red neuronal profunda usada para las tareas de visión por computadora, como la detección de objetos en imágenes.</p>	<p>Se usa el aprendizaje profundo como TensorFlow o PyTorch.</p>	<p>Depende del tamaño de modelo y recursos del hardware.</p>	<p>Depende de la Visión por computadora, y detección de objetos.</p>	<p>Depende de su uso, pero este tipo puede manejar datos en tareas de visión.</p>	<p>Se considera baja debido a su complejidad</p>	<p>Capacidad de aprendizaje de patrones no lineales en imágenes y datos similares.</p>
RNN		<p>Recurrent Neural Network también son una clase de redes</p>	<p>Uso de aprendizaje profundo</p>	<p>Depende del tamaño de modelo y recursos del hardware.</p>	<p>Precisión alta en datos secuenciales en análisis como texto o</p>	<p>Alta capacidad de la dimensión de datos en tareas secuenciales.</p>	<p>Su interoperabilidad es baja debido a su naturaleza o complejidad.</p>	<p>Aprende patrones no lineales en secuencias de datos.</p>

neuronales, su
uso es en el
modelado de
secuencias de
datos.

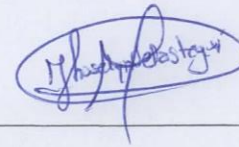
series
temporales.

Revisor por:



Director
Msc. Fabián Geovanny Cuzme Rodríguez

Elaborado por:



Estudiante
Jhoselyn Lizeth Velastegui Morales