EDUCACIÓN
SECRETARÍA DE EDUCACIÓN PÚBLICA

TECNOLÓGICO NACIONAL DE MÉXICO

Tecnológico Nacional de México

Centro Nacional de Investigación y Desarrollo Tecnológico

Tesis de Doctorado

Seguridad Inteligente en la red del Internet de las cosas

presentada por

MC. Eddy Alberto Pola Jiménez

como requisito para la obtención del grado de

Doctor en Ciencias de la Computación

Director de tesis

Dr. Máximo López Sánchez

Codirector de tesis

Dr. Juan Gabriel González Serna

Cuernavaca, Morelos, México. Agosto de 2024.

cenidet	ACEPTACIÓN DE IMPRESIÓN DEL DOCUMENTO DE TESIS DOCTORAL	Código: CENIDET-AC-006-D20
		Revisión: O
	Referencia a la Norma ISO 9001:2008 7.1, 7.2.1, 7.5.1, 7.6, 8.1, 8.2.4	Página 1 de 1

Cuernavaca, Mor., a 06 de junio de 2025

DR. CARLOS MANUEL ASTORGA ZARAGOZA
SUBDIRECTOR ACADÉMICO
P R E S E N T E

AT'n: DR. JUAN GABRIEL GONZÁLEZ SERNA
PRESIDENTE DEL CLAUSTRO DOCTORAL DEL
DEPARTAMENTO DE CIENCIAS COMPUTACIONALES

Los abajo firmantes, miembros del Comité Tutorial del estudiante **M.C. Eddy Alberto Pola Jiménez** *manifiestan que después de haber revisado el documento de tesis titulado* **"Seguridad Inteligente en la Red del Internet de las Cosas"**, *realizado bajo la dirección del* **Dr. Máximo López Sánchez** *y codirección del* **Dr. Juan Gabriel González Serna**, *el trabajo se ACEPTA para proceder a su impresión.*

A T E N T A M E N T E
Excelencia en Educación Tecnológica®
"Conocimiento y Tecnología al Servicio de México"

DR. MÁXIMO LÓPEZ SÁNCHEZ
TECNM/CENIDET

DR. JUAN GABRIEL GONZÁLEZ SERNA
TECNM/CENIDET

DR. DANTE MÚJICA VARGAS
TECNM/CENIDET

DR. NIMROD GONZÁLEZ FRANCO
TECNM/CENIDET

DR. JOSÉ ALEJANDRO REYES ORTIZ
UAM/AZCAPOTZALCO

c.c.p: C Verónica Sotelo Boyas/ Jefa del Departamento de Servicios Escolares
c.c.p: C Nile Alejandro Castro Sánchez / Jefe del Departamento de Ciencias Computacionales
c.c.p: Expediente

Educación
Secretaría de Educación Pública

Tecnológico Nacional de México

Centro Nacional de Investigación y Desarrollo Tecnológico
Subdirección Académica

Cuernavaca Mor, 06/junio/2025
Oficio No. SAC/133/2025
Asunto: Autorización de impresión de tesis

EDDY ALBERTO POLA JIMÉNEZ
CANDIDATO AL GRADO DE DOCTOR
EN CIENCIAS DE LA COMPUTACIÓN
P R E S E N T E

Por este conducto, tengo el agrado de comunicarle que el Comité Tutorial asignado a su trabajo de tesis titulado **"Seguridad Inteligente en la Red del Internet de las Cosas"**, ha informado a esta Subdirección Académica, que están de acuerdo con el trabajo presentado. Por lo anterior, se le autoriza a que proceda con la impresión definitiva de su trabajo de tesis.

Esperando que el logro del mismo sea acorde con sus aspiraciones profesionales, reciba un cordial saludo.

A T E N T A M E N T E
Excelencia en Educación Tecnológica®
"Conocimiento y Tecnología al Servicio de México"

EDUCACIÓN
CENTRO NACIONAL DE INVESTIGACIÓN Y DESARROLLO TECNOLÓGICO
SUBDIRECCIÓN ACADÉMICA

CARLOS MANUEL ASTORGA ZARAGOZA
SUBDIRECTOR ACADÉMICO

c.c.p. Departamento de Ciencias Computacionales
Departamento de Servicios Escolares

CMAZ/lmz

2025 Año de La Mujer Indígena

Interior Internado Palmira S/N, Col. Palmira,
C. P. 62490, Cuernavaca, Morelos Tel. 01 (777) 3627770, ext. 4104,
e-mail: acad_cenidet@tecnm.mx tecnm.mx | cenidet.tecnm.mx

cenidet

Dedicatoria

A la burocracia eterna,
al estrés crónico disfrazado de productividad,
y a ese matutino dulce aroma a burnout con café recalentado.
A todos los formularios, correcciones absurdas
y noches donde lo único claro era la incertidumbre.
Gracias por nada y lo siento, logré terminar.

Agradecimiento

Primeramente, gracias a Dios, quien día a día me confirmó el transitar de este camino, donde la ciencia y la fe se arraigan para crear una convicción inquebrantable del impacto que cada uno de nosotros puede trascender aportando conocimiento y experiencia.

También, deseo agradecer al Consejo Nacional de Ciencia y Tecnología (CONACYT) por esta tremenda posibilidad de crecimiento que me brindó a través de la manutención de su beca. Este es un logro que sin su ayuda no habría sido posible conseguir en el tiempo previsto.

La realización de esta tesis fue un esfuerzo en conjunto con mi director de tesis, el Dr. Máximo López Sánchez y mi codirector de tesis, Dr. Juan Gabriel González Serna, por eso expreso mi más profundo agradecimiento a ellos y al Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET) por haberme abierto las puertas a esta grandiosa oportunidad.

Agradezco a mi familia, pues pase lo que pase, y sea quien yo sea, siempre estarán ahí para apoyarme sin importar el camino que yo decida tomar.

Finalmente, en el mismo grado de importancia, a los amigos (así como los que son más que amigos) que han estado animándome, apoyándome y permitiéndome crecer con sus vidas. Rodearme de todos ellos ha sido la más grata experiencia que haya tenido en mi vida.

Resumen

En la última década, el Internet de las Cosas (IoT) ha revolucionado la forma en que los objetos se conectan e interactúan, permitiendo su aplicación en diversos sectores como la salud, agricultura, industria y transporte. Entre estos avances destaca la implementación de vehículos no tripulados, o drones, que gracias a la integración del IoT han ganado protagonismo en múltiples industrias. Sin embargo, esta interconectividad también plantea desafíos significativos en términos de seguridad informática. En particular, los ataques de denegación de servicio (DoS) representan una amenaza crítica que puede afectar el funcionamiento de estos sistemas. Esta tesis analiza dichos riesgos y propone un nuevo modelo de detección de ataques DoS y LDoS mediante tormentas de paquetes, alcanzando una precisión de detección entre el 94% y el 100%. Como resultado, este trabajo culmina con la publicación de un artículo indexado en el tercer cuartil del Journal Citation Reports (JCR).

Resume

Over the last decade, the Internet of Things (IoT) has transformed the way everyday objects connect and communicate, enabling applications across various sectors such as healthcare, agriculture, industry, and transportation. Among these innovations, unmanned vehicles—commonly known as drones—have become increasingly important, powered by IoT integration. However, this widespread connectivity introduces significant cybersecurity challenges. One of the most critical threats is the Denial of Service (DoS) attack, which can compromise system functionality. This thesis explores these risks and presents a novel detection model for DoS and LDoS attacks through packet storm analysis, achieving a detection accuracy between 94% and 100%. The work concludes with the publication of a JCR-indexed article positioned in the third quartile.

ÍNDICE GENERAL

Índice de tablas

Índice de Ilustraciones

CAPÍTULO 1. INTRODUCCIÓN

En la última década, el avance tecnológico ha llevado a una revolución en la forma en que interactuamos con la tecnología, transformando la manera en que los objetos cotidianos se conectan y comunican entre sí. En este contexto, el Internet de las Cosas (IoT) ha emergido como una fuerza impulsora detrás de esta transformación, extendiendo la conectividad más allá de los dispositivos convencionales a una red global de objetos interconectados. La interconexión de objetos tan comunes como electrodomésticos, hasta equipos especializados como monitores de frecuencia cardiaca, han permitido al IoT ubicarse en campos como educación, hogar, salud, agricultura, industria, entre otros.

Hablando del mercado, en 2018 se invirtió 646 mmdd a nivel mundial y se estima que para este año (2023) hayan más de 1,300 mmdd invertidos (Fernández, 2023). Si se habla de crecimiento, datos estadísticos del 2017 mostraron una proyección de 20.35 mil millones de dispositivos IoT interconectados a 35.82 mil millones para este año a nivel mundial (Statista, 2023). Un área de aplicación que ha experimentado un crecimiento exponencial en el contexto del IoT es la implementación de vehículos no tripulados.

Los vehículos no tripulados, comúnmente conocidos como drones, han pasado de ser juguetes tecnológicos a desempeñar roles críticos en diversas industrias, desde la agricultura hasta la entrega de bienes. Este fenómeno ha sido posible gracias a la integración del IoT en el diseño y funcionamiento de estos vehículos, permitiendo una comunicación instantánea y eficiente entre ellos y su entorno.

Sin embargo, a medida que la adopción del IoT en vehículos no tripulados aumenta, surge una preocupación crucial: la seguridad informática. La interconexión masiva de dispositivos en el IoT presenta desafíos significativos, especialmente en términos de proteger la integridad, confidencialidad y disponibilidad de los datos transmitidos. Según datos de Cisco(2018), los adversarios utilizan métodos cada vez más sofisticados y se ayudan de tecnologías válidas como cómputo en la nube para explotar las vulnerabilidades de IoT. En este contexto, los ataques de denegación de servicio (DoS) emergen como amenazas potenciales que podrían comprometer la funcionalidad y la seguridad de los vehículos no tripulados.

Este trabajo explora la intersección crítica entre el uso del IoT en vehículos no tripulados y la seguridad informática, centrándose especialmente en los riesgos asociados con los ataques de denegación de servicio por tormenta de paquetes. A través de un análisis exhaustivo, se encuentra lo siguiente: se puntualizan las variables existentes que identifican los ataques DoS en diferentes aplicaciones, se plantea la efectividad de dichas variables en este campo del IoT. El aporte de este tema de tesis consiste en la propuesta de un nuevo modelo de detección de ataques DoS y LDoS exponiendo un rendimiento en la identificación que va desde el 94% hasta el 100% de precisión, culminando en la publicación de un artículo JCR ubicado en el tercer cuartil.

1.1 Descripción del problema

Un sistema de detección inteligente para denegación de servicio podría tener un alto índice de evaluación, pero representa un potencial y costoso peligro si no se logra de manera oportuna. Hace falta conocer el tiempo de detección de un ataque DoS para dimensionar la afectación real a un nodo final y además conocer la efectividad, en este enfoque, de un sistema de detección de intrusiones.

Por lo tanto, el problema se describe de la siguiente manera: En un enfoque descentralizado, **cómo caracterizar la información que permita identificar, detectar y prevenir una intrusión antes de que el nodo atacado sea parcial o completamente afectado**.

1.1.1. Hipótesis

- Variables asociadas al uso de CPU y RAM influirán en el grado de detección de ataques DoS a un nodo final.

1.2 Objetivos

1.2.1. Objetivo general

Caracterizar la información proveniente de un dispositivo final que permita identificar un ataque DoS en redes multisalto y prevenirlo antes de afectar al nodo.

1.2.2. Objetivos Específicos

- Experimentación con conjuntos de datos estudiados en el estado del arte.
- Montar una red multisalto.
- Caracterizar la información en redes multisalto.
- Experimentar un trabajo publicado sobre una red multisalto para determinar tiempo y tasa de detección.
- Proponer un método o técnica para la recolección de datos de nodos finales en una red multisalto.
- Diseñar un algoritmo para la detección de intrusiones con un sistema de prevención.
- Realizar pruebas de rendimiento para detección de intrusiones.

CAPÍTULO 2. MARCO TEÓRICO

2.1 Internet de las cosas

El internet de las cosas (*IoT* por sus siglas en inglés) se ha vuelto una tendencia en la última década, especialmente en el ámbito de la investigación.

Antes de dar una definición para esta tesis, se plantean diferentes definiciones de entidades importantes en la era de la tecnología; Por ejemplo, RedHat (2023) define al IoT como un *proceso* donde dispositivos físicos se interconectan para compartir datos o enviarse instrucciones.

Por su parte, Kaspersky (s.f.) lo define como un *sistema* "*… de dispositivos electrónicos interconectados que puede recopilar y transferir datos a través de una red inalámbrica sin intervención de personas*".

Hewlett-Packard-Enterprise (s.f.) coincide con los diferentes actores que se describen en las anteriores definiciones, a diferencia que HP® define al IoT como una *red creciente* y que los dispositivos interconectados no son *tradicionales* de la computación.

Definimos entonces al internet de las cosas como una red de objetos cotidianos inteligentes que envían, reciben y gestionan información a través de un protocolo dado. Aunque la manera en que se conectan es de manera inalámbrica, muchas veces dicha red es, pocas veces, con la ayuda de la intervención humana.

Hoy en día, el IoT está presente en tareas como medicina electrónica (E-health por sus siglas en inglés), agricultura, educación, industria 4.0, gobierno, hogares inteligentes, construcciones inteligentes, etc. Todos estos contextos de aplicación surgen de la necesidad de recolectar datos del mundo físico a través de sensores y hacer uso dependiendo de los objetivos, con esto nacen las redes mobiles adhoc (MANET por sus siglas en inglés).

2.1.1. Manet, Vanet y Fanet

Se define a las MANETs como redes de sensores inalámbricos interconectados entre ellos y que tienen la capacidad de auto configurarse y auto gestionarse cuando sea necesario. Se desempeñan en espacios ciber-físicos y tiene la finalidad de compartir información de sensores entre estos mismos sin la necesidad de que exista una infraestructura de red propia.

Una característica muy importante es que los dispositivos están en constante movimiento en un espacio físico. Dadas las condiciones de un dinamismo en la ubicación de cada sensor, indicaría un cambio circunstancial en las condiciones de su conexión y envío/recepción de datos, esta es la finalidad de que puedan ser auto organizados.

Las redes de vehículos adhoc y redes de vehículos voladores adhoc (VANET y FANET por sus siglas en inglés) son redes de vehículos no tripulados que buscan ser auto organizables. Aunque también se incorporan con muchos sensores para recolección de datos (dependiendo del contexto de aplicación), una de sus tareas más comunes es la de compartir imágenes en tiempo real para uso de búsquedas y ubicación.

Aunque los objetos en las redes MANET son sensores casi en su totalidad, las VANETs y FANETs tienen capacidades de CPU, RAM, RED y almacenamiento más demandantes y, por consecuencia, con una configuración más poderosa.

Las redes VANET y FANET se utilizan con redes WiFi y LTE, pero por la característica y necesidad donde no siempre se cuenta con una infraestructura propia, comienzan a surgir alternativas de conexión como zigbee.

2.2 Tecnología zigbee

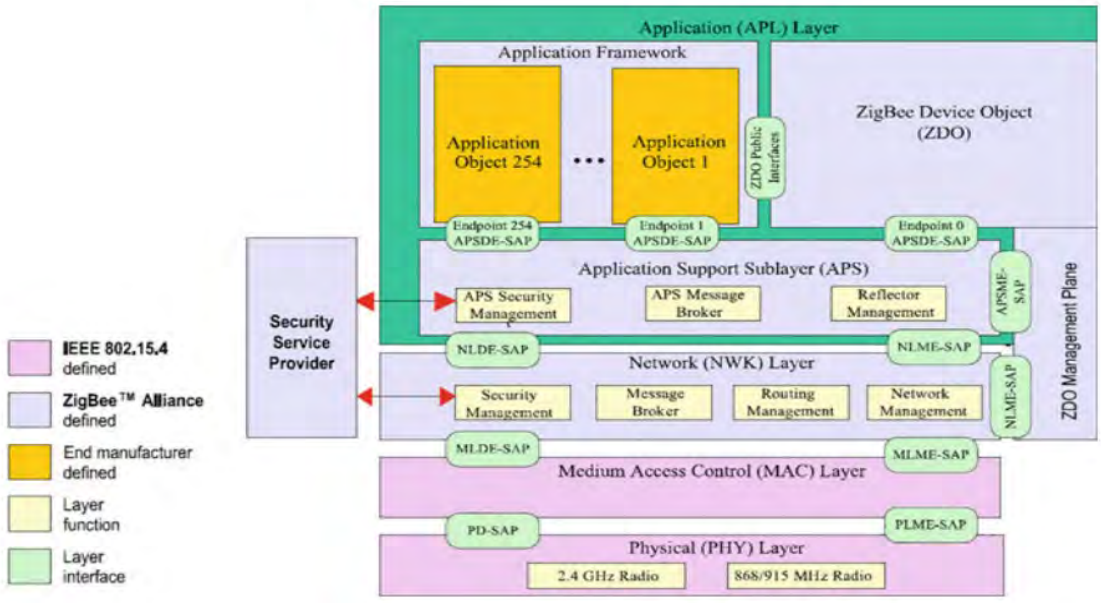
Zigbee es el estándar de un protocolo para redes inalámbricas de baja velocidad y bajo consumo utilizado por aplicaciones de bajo costo, bajo consumo y baja velocidad. Los dispositivos Zigbee trabajan en las bandas de frecuencias de 868 MHz, 915 MHz y 2.4 GHz con una velocidad máxima de transmisión de 250 Kb/s. En aplicaciones típicas como hogares inteligentes y redes de sensores, los dispositivos zigbee trabajan en modo bajo consumo de energía o en estado de sueño la mayoría de las veces, por eso sus baterías pueden durar años.

El estándar Zigbee fue formulado por la *Zigbee Alliance* la cual es una composición de cientos de compañías en semiconductores, desarrollo de software e industrias manufactureras de dispositivos.

Las aplicaciones típicas de Zigbee incluyen redes de sensores, control automático y advertencia de desastres naturales. Estos dispositivos trabajan en estado de sueño casi siempre y solo se comunican ocasionalmente. El protocolo es caracterizado por tener una transmisión de datos confiable, tamaño pequeño de sistemas y bajo requerimientos de recursos. A diferencia de Bluetooth, Zigbee ha adoptado una topología de red multisalto autorganizada.

2.2.1. La relación entre Zigbee e IEEE 802.15.4

Tal y como se aprecia, el estándar IEEE 802.15.4 define la capa física y la capa de acceso al medio. Dicho estándar fue formulado por el comité de estándares IEEE 802. Sin embargo, el protocolo de red Zigbee define únicamente la capa de red, capa de aplicación y capa de seguridad. Aunque la capa física y de acceso al medio son definidas por el estándar IEEE 802.15.4, estos aún son parte del protocolo Zigbee.



[1]

La relación entre estos dos estándares se explica con la fig. 1 representada con el modelo OSI clásico de capas.

2.2.2. Estructura de las tramas 802.15.4

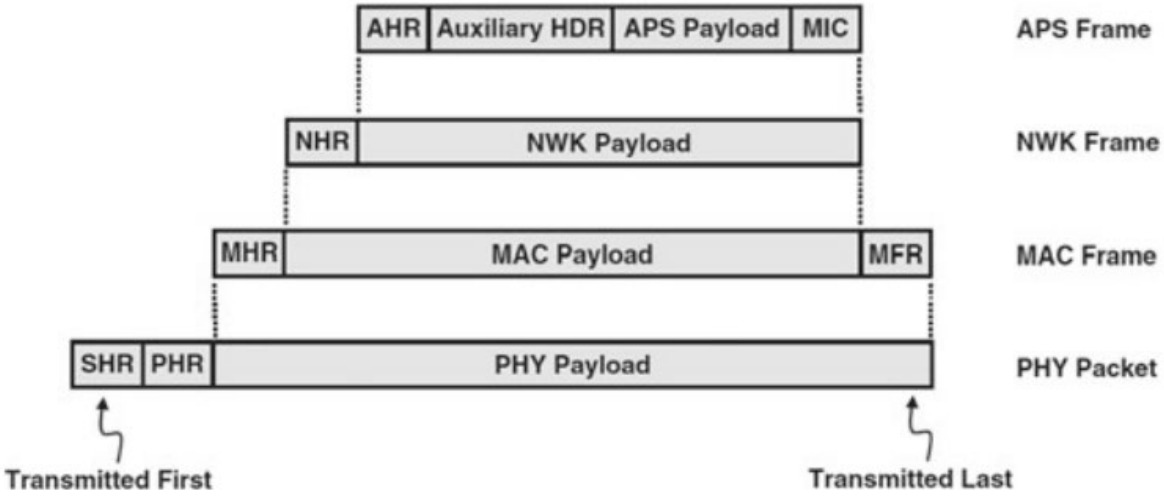
Como se sabe, las capa física y capa de acceso al medio (*MAC* por sus siglas en inglés) son definidas por el estándar 802.15.4, pero en lugar de ello se refieren como parte de Zigbee[2].

La capa física de Zigbee puede trabajar en las bandas de frecuencia de 868 MHz, 915 MHz y 2.4 GHz, los cuales pueden ser divididos en 27 canales. Similar al estándar 802.11, Zigbee usa la tecnología de espectro ampliado de secuencia directa (*DSSS* por sus siglas en inglés). Si no se reconfigura, la comunicación permanecerá en el mismo canal sin hacer saltos. Además, capturar datos en una red Zigbee es más fácil que en Bluetooth.

Los datos y los comandos son transmitidos vía tramas de datos. Los datos en la capa de aplicación son empaquetados capa tras capa y, finalmente, transmitido por la capa física hacia el nodo destino.

[1] (Yang & Huang, 2018)

[2] La capa física y capa de acceso al medio de 802.15.4 también puede ser llamada capa física y capa de aaceso al medio de Zigbee.



[3]

Como se puede apreciar en la Fig. 2 ,los datos Zigbee son empaquetados capa tras capa desde la capa superior hacia la inferior durante la transmisión, con la cabecera correspondiente y el control de información agregado en cada capa. En la recepción, los datos son desempaquetados justamente de manera opuesta a como se hizo en la transmisión, es decir, desde la capa inferior hacia la superior.

Los siguientes puntos son una interpretación fiel de cómo se estructura cada trama de datos por capa:

La trama de datos en la capa física: En esta trama, la cabecera de sincronización (*SHR* por sus siglas en inglés) sincroniza al transmisor y receptor, en este orden, para enlazar el flujo de datos. La cabecera de la capa física (*PHR* por sus siglas en inglés) contiene la longitud de la trama y la carga útil de la capa física (*PHY payload*) es provista por las capas superiores y contiene los datos y comandos que serán transmitidos.

La trama de datos MAC: Esta trama es transmitida como carga útil de la capa física y consiste en tres partes – cabecera MAC (*MHR* por sus siglas en inglés), carga útil MAC (*Payload MAC* por sus siglas en inglés) y cola MAC (*MFR por sus siglas en inlés*); de entre los cuales, MHR contiene la información de direccionamiento y seguridad de la capa MAC, *MAC Payload* contiene la trama de datos de la capa de red y *MFR* contiene un control de redundancia cíclica de 16 bits.

La trama de la capa de red: Esta trama consiste de una cabecera (*NHR*) y una carga útil (*NWK Payload*). El primero contiene el direccionamiento y la seguridad de la capa de red, el segundo contiene la trama de la sub-capa de soporte de aplicación (*APS*).

[3] (Yang & Huang, 2018)

En la trama *APS*, la cabecera *AHR* contiene el direccionamiento y el control de información de la capa de información; La cabecera auxiliar (*Auxiliary HDR*) contiene información de seguridad que incluye la llave del número de serie.

También se podría agregar seguridad adicional a la capa MAC y la capa de red agregando una cabecera auxiliar en las respectivas capas. La carga útil de las subcapas de soporte de aplicación (*APS Payload*) contienen datos de aplicación o comandos. El código de integridad de mensaje (*MIC*) es utilizado para checar si los mensajes han sido manipulados. La generación de los *MIC* se lleva a cabo con el uso de una llave. De esta manera, todas las actividades de manipulación que no usen dicha llave serán descubiertos.

2.2.3. Las diferentes tramas MAC en Zigbee

A diferencia de otros protocolos inalámbricos como WiFi, Zigbee solo utiliza las siguientes tramas en la capa MAC para transmitir datos:

Trama *Beacon*: Esta trama es utilizada en el escaneo de la red. Cuando un nuevo nodo intenta unirse a una red, primero enviará una consulta *beacon*. Los nodos que han recibido la consulta *beacon* enviarán su propio *beacon* para que el nuevo nodo pueda encontrar la red.

Trama de datos: Esta trama es usada para intercambiar todo tipo de datos con una longitud máxima de 114 bytes.

Trama de reconocimiento[4]: El emisor podría requerir que el receptor responda con una trama de reconocimiento para indicarle la recepción exitosa de los datos.

Trama de comando: Similar a la trama de administración de red en el estándar 802.11, la trama de comando maneja la asociación, disociación, conflictos de dirección de red y transmisión de datos caché.

2.2.4. Tipos de dispositivos y topología de red Zigbee

En una red IEEE 802.15.4 los dispositivos son clasificados dentro de dos tipos basados en sus capacidades: dispositivos de función completa (*FFD* por sus siglas en ingés) y dispositivos de función reducida (*RFD* por sus siglas en inglés).

Los *FFD*s pueden completar todas las tareas especificadas en el estándar 802.15.4, mientras que los *RFD*s solamente pueden realizar ciertas funciones. Por ejemplo, los *FFD*s se pueden comunicar con todos los dispositivos en la red y los *RFD*s solamente se pueden comunicar con los *FFD*s. Los *RFD*s son utilizados en aplicaciones simples, como switches.

Además, los dispositivos en una red IEEE 802.15.4 pueden ser clasificados en 3 tipos basados en sus roles: Coordinador PAN (Del inglés *Personal Area Network*), coordinador y dispositivo ordinario. Los coordinadores son *FFD*s en los cuales retransmiten la información de la red. Si un coordinador también funciona como el controlador principal de una red de área personal, entonces es llamado coordinador PAN.

[4] Conocidas como tramas *Acknowledge*.

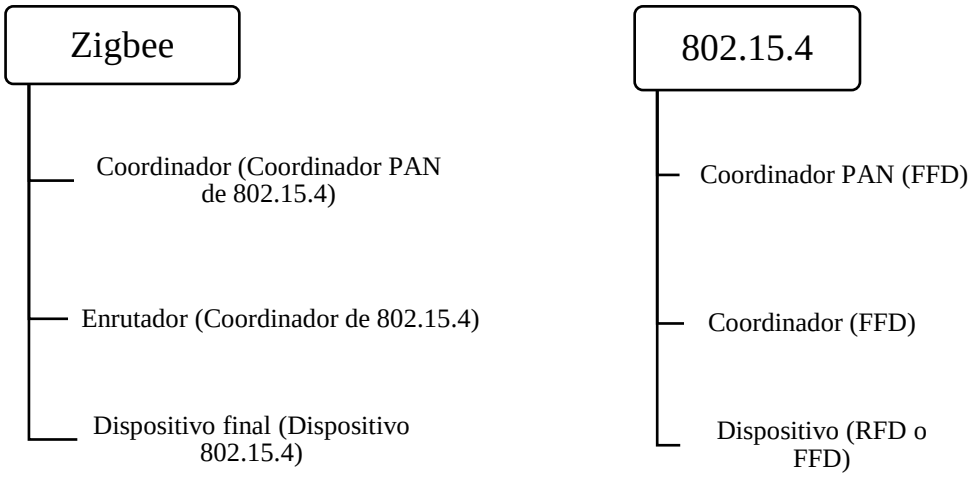


Fig. 3. Similitudes en los roles 802.15.4 y Zigbee.

En la terminología Zigbee, los mismos roles pueden tener diferentes nombres. Los coordinadores, enrutadores y dispositivos finales corresponden respectivamente a los coordinadores PAN, coordinadores y dispositivos ordinarios en el estándar 802.15.4, tal como se aprecia en la Fig. 3 .

Algunas de las definiciones de los roles de los dispositivos Zigbee son los siguientes:

Coordinador: es un FFD responsable de controlar la red entera, retransmitiendo los mensajes y autenticación de los nuevos nodos (incluyendo la denegación de su acceso).

Ruteador: es un FFD responsable de transmitir y reenviar paquetes de datos. Se puede comunicar con el coordinador y el dispositivo final.

Dispositivo final: Es un RFD el cual no puede reenviar datos ni comunicarse con el dispositivo final. Solamente se puede comunicar con el enrutador o el coordinador.

Una red Zigbee es gestionada por la capa de red e implementada en tres estructuras topológicas: estrella, malla y árbol. Aunque cada red necesita un coordinador administrador, algunos enrutadores también se requieren para enviar y reenviar datos dependiendo de la topología.

2.2.5. Redes Zigbee

Las redes Zigbee son implementadas por la capa de red (Capa *NWK* por sus siglas en inglés), la cual es también responsable del descubrimiento de dispositivos, asignación de direcciones de red y ruteo. El proceso de red es el siguiente:

Un coordinador (también FFD) escanea todas las redes que operan en el canal especificado a través del descubrimiento de dispositivos (utilizando la trama de beacons); entonces, aleatoriamente se selecciona una dirección de red diferente a las existentes. Finalmente, este recibirá consultas de acceso enviadas por los enrutadores y dispositivos

finales. Cuando un nodo accede a la red, el coordinador le asignará una dirección de red de 16 bits.

2.2.6. Capa de aplicación de Zigbee

La capa de aplicación es la capa más alta especificada en el estándar Zigbee e incluye la interfaz de operación de los objetos de aplicación Zigbee, los cuales son definidos por la Zigbee Alliance o los manufactureros de los productos Zigbee.

Cada dispositivo Zigbee debe implementar un objeto de dispositivo Zigbee (ZDO). El ZDO tiene la capacidad de configurar el rol del dispositivo (coordinador, ruteador o dispositivo final), proveer servicios de seguridad (como la configuración y borrado de llaves) y administrar la red (como las asociaciones y disociaciones de nodos). El ZDO también ha definido un perfil de dispositivo Zigbee unido a un punto final de aplicación numerado con 0.

2.2.7. La subcapa de soporte de aplicación de Zigbee

La subcapa de soporte de aplicación (APS) provee una interfaz entre la capa de aplicación y la capa de red a través de un conjunto de servicios universales, los cuales pueden ser usados por ZDOs y objetos de aplicación personalizados por los fabricantes. La APS provee estos servicios a través de entidades de datos (APSDE) y entidades administrativas (APSME). Las APSDE proveen transmisión de datos a través de APSDE-SAP y APSME proveen la administración del servicio a través de APSME-SAP y mantienen una base de datos de objetos administrados llamados información base APS (AIB).

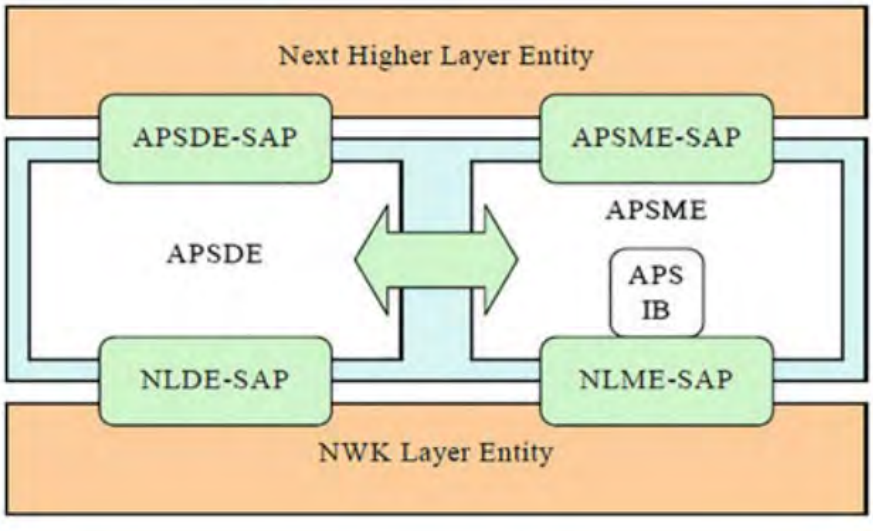


Fig. 4. Interfaces involucradas en la interacción de la capa de red y capa de aplicación del estándar Zigbee.[5]

[5] (Yang & Huang, 2018)

Así como se ilustra en la Fig. 4, las entidades de datos de la subcapa de soporte de aplicación (APSDE) proveen los siguientes servicios a través de sus correspondientes puntos de acceso (APSDE-SAP):

Generación de paquetes de datos: los APSDEs agregan una cabecera de protocolo a los paquetes de datos de la capa de aplicación (PDU) para generar paquetes de datos de la subcapa de soporte de aplicación.

Unión: Dos dispositivos unidos el uno con el otro se pueden comunicar de manera bidireccional.

Filtrado de direcciones multicast: los APSDEs pueden filtrar paquetes de datos dirigidos a direcciones multicast según las propiedades de grupo de los puntos finales de aplicación.

Transmisión confiable de datos: los APSDE pueden mejorar más la confiabilidad de la transmisión de datos basados en la capa de red a través de mecanismos de retransmisión punto a punto.

2.2.8. Filtrado de datos duplicados.

Segmentación de paquetes: si un mensaje se ha extendido de la longitud máxima transmisible por la capa de red, los APSDEs pueden dividir un paquete de datos en varios segmentos. Por otro lado, estos también pueden ensamblar los datos segmentados en un mensaje completo.

Las APSME proveen una interfaz interactiva entre las pilas del protocolo para la aplicación. Los servicios provistos por APSME a través de su correspondiente servicio de punto de acceso (APSME-SAP) incluye:

Administración de enlaces: permite emparejar dos dispositivos basados en sus servicios y requerimientos.

Administración AIB: permiten leer y modificar las propiedades internas del AIB.

Seguridad: Permiten la autenticación (vía clave) y la conexión con otros dispositivos.

Administración de grupo: permite el direccionamiento de múltiples dispositivos con solo una dirección (direccionamiento multicast) y agregar o eliminar dispositivos del grupo.

2.2.9. Perfil de aplicación en Zigbee

El perfil de aplicación define el formato de los mensajes y el proceso para su manejo. Además, las entidades de la aplicación ejecutándose en diferentes dispositivos pueden interactuar entre ellos en tanto que ellos cumplan con el mismo perfil. Por ejemplo, una casa inteligente provee un perfil unificado que le permite a los dispositivos interactuar entre ellos sin importar los fabricantes.

2.2.10. Seguridad Zigbee

Zigbee ha adoptado un algoritmo de encriptación AES de 128 bits y su seguridad depende del almacenamiento de llaves simétricas, el mecanismo de protección, mecanismos de encriptación e implementación de estrategias relevantes. Además, la arquitectura de la seguridad en Zigbee es, en esencia, la pre distribución, inicialización, uso y almacenamiento de claves.

Tipos de llaves

Zigbee define 3 tipos de llaves: La llave maestra, la llave de red y la llave de enlace.

Llave maestra: La llave maestra ha sido guardada en los dispositivos desde su despliegue y es utilizada para proteger la llave del enlace durante el establecimiento de llave simétrica (SKKE).

Llave de red: La llave de red de 128 bits es compartida a todos los nodos dentro de la red para encriptar datos multicast y broadcast. La llave de red podría ser transmitida en texto plano durante el acceso de un nodo.

Llave de enlace: Es compartida por cada dos nodos para encriptar su comunicación. Esta llave es administrada la capa de aplicación, pero permanece casi sin uso en la práctica.

Autenticación para nodos Zigbee

Las incorporaciones de los nuevos nodos a la red son autenticadas por los siguientes tres métodos:

Listas de control de acceso (ACL): Los nodos en la red autentican uno a otro basados en su dirección MAC (dirección física), y cada nodo mantiene un listado de direcciones MAC de los nodos con los cuales se puede comunicar. Este método es solo aplicable cuando se combina con la función de combinación de datos CCM* (Con una función opcional de encriptación de datos), porque un atacante puede falsificar la dirección MAC si no se autentica con CCM*.

En el modo de seguridad/residencial estándar, un nodo debe ser autorizado por el centro de confianza enviando la llave de red antes de acceder a la red (Se permite o deniega el acceso dependiendo de las direcciones MAC de los nodos). Si la llave de red está preguardada en el nodo, el centro de confianza le enviará una llave *full-zero* que significa su autenticación, de otra manera el centro de confianza enviará la llave en texto plano al nuevo nodo (nótese que este proceso es muy riesgoso). Además, en este modo de seguridad, el nodo que solicita acceso recibe la llave transmitida por el centro de confianza si autenticar el centro de confianza.

En el modo de seguridad/residencial avanzado, la transmisión de texto plano de la llave de red no está permitido. Cuando un nodo requiere acceso a la red, necesita generar (nótese ‘generar’) la llave de red utilizando el establecimiento de llaves simétricas (SKKE).

Si el nodo no tiene la llave maestra, el centro de confianza puede enviar la llave en texto plano a este.

2.2.11. Ataques Zigbee

Herramientas de ataque

CC2531 USB Dongle

Esta herramienta fue desarrollada por Texas Instruments basado en CC231 y es similar a un dispositivo Zigbee de función completa. El Dongle puede ser programado para volverse un nodo en una red Zigbee y su firmware incorporado es capaz de capturar paquete de datos y es compatible con muchas piezas del protocolo de análisis software.

Depurador CC

Un depurador CC es utilizado para depurar chips SoC RF de bajo consumo producido por Texas Instruments y puede ser utilizado con una plataforma de desarrollo embebido IAR que depura chips Zigbee y descargan su software. El depurador también podría ser utilizado con el Programador Flash SmartRF para leer el dispositivo firmware.

Unicorn_Zigbee

Este es un módulo desarrollado por el 360UnicornTeam basado en el chip de tecnologías de información CC2530. Este nodo puede ser utilizado para replicar datos y actuar como nodo malicioso.

Ambiente de desarrollo integrado IAR

Este es utilizado para desarrollar aplicaciones Zigbee. Después del análisis al sistema designado, se puede desarrollar una aplicación atacante sobre IAR basada en Unicorn_Zigbee.

2.3 Sistema de detección de intrusiones

Se define como una intrusión a cualquier tipo de actividad no autorizada que tiene la finalidad de dañar un sistema de información.

Un sistema de detección de intrusiones (IDS por sus siglas en inglés) es un sistema de hardware o software que determina acciones maliciosas en sistemas de cómputo con la finalidad de mantener una seguridad provista. La meta de un IDS es identificar distintos comportamientos anómalos en el uso de red o computación que un firewall tradicional no podría.

Los IDS pueden ser categorizados en dos grandes grupos: IDS basados en firmas (SIDS por sus siglas en inglés) e IDS basadas en anomalías (AIDS por sus siglas en inglés). El primero tiene que ver su uso con programas de antivirus en red, haciendo alusión a un registro en una base de datos de ataques reconocidos, mientras que el segundo normalmente involucra con un análisis de comportamiento interpretando y reconociendo patrones de los ataques.

Además, existen tres grandes enfoques de detección y que se relacionan muy de cerca con los AIDS, estos son:

Sistema de detección de intrusiones basado en host (HIDS por sus siglas en inglés): Por lo general, son programas que se instalan en los dispositivos a salvaguardar y desde ahí se aplican análisis por reglas, firmas o anomalías para la identificación de intrusiones.

Sistema de detección de intrusiones basados en red (NIDS por sus siglas en inglés): Éstos sistemas se centran en el análisis de tráfico de red y cada uno de sus componentes para hacer detección de comportamientos anómalas.

Sistema de detección de intrusiones híbrido (HYIDS por sus siglas en inglés): Es una combinación de los dos enfoques anteriores.

CAPÍTULO 3. CAPÍTULO III. ESTADO DEL ARTE

3.1 Trabajos recientes

La búsqueda de artículos del estado del arte se centra en el título de esta tesis doctoral que comenzó con el nombre de '*Seguridad en la red del Internet de las cosas*'. Los estudios e investigaciones, a razón de '*surveys*', señalan la brecha que existe en la ciberseguridad del internet de las cosas y en todos los documentos se exponen los ataques y algunas contramedidas que son tomadas por los encargados de seguridad de la red. Sin embargo, surgió las primeras preguntas de esta investigación: De todos los contextos existentes en el IoT, ¿Se puede aseverar que cada ataque aplica para cada aplicación de IoT? Es decir, dada la naturaleza de la conectividad y tipos de dispositivos utilizados en cada contexto de IoT, ¿Podría haber ataques que no tuvieran efecto o no fuesen posible hacer para algún contexto en específico? Todo esto resultó en una recopilación de trabajos que exponen de manera teórico-práctica la vulnerabilidad de diferentes dispositivos IoT aplicados a diferentes contextos.

Tabla 1. Documentos pragmáticos que prueban la vulnerabilidad de IoT.

Trabajo	Año	Tipo	Respecto a vulnerabilidades
Lu & Xu (2019)	2019	Journal	Presentación de una taxonomía de ataques en el IoT. Presentación de esquemas de seguridad en IoT. Exposición de temas tendencia para investigación de ciberseguridad.
Siboni, Shabtai, & Elovici (2018)	2018	Conference	Exposición pragmática de la vulnerabilidad de un dispositivo IoT Bring Your Own Device (BYOD). Las pruebas indican un caso de uso al transportar un smartwatch infectado con malware a una red empresarial.

Tweneboah-Koduah, Skouby, & Tadayoni (2017)	2017	Journal	Exposición de una base de datos (2015) con los ataques registrados al IoT. Exposición pragmática de la vulnerabilidad de dispositivos IoT en redes Smart Home. Pruebas de vulnerabilidad en el caso de uso Smart Metering con ataques de inyección SQL y DoS.
Kolias, Kambourakis, Stavrou, & Voas (2017)	2017	Magazine	Presentación de dos casos reales que realizaron ataques DDoS en el 2016 en el que alcanzaron tráfico de hasta 620 Gbps. Exposición del comportamiento de botnets (dispositivos IoT infectados) culpables de realizar el ataque DDoS. Reporte de más de 493,000 variaciones del malware Mirai.
Wood, Apthorpe, & Feamster (2018)	2017	Conference	Se examinó una transmisión de datos de texto claro en el caso de uso de IoT médico. Se estudiaron cuatro casos de los cuales uno de ellos reveló información sensible de la salud de un paciente.
Fuller, Jenkins, & Tjølsen (2017)	2017	Document	Documento realizado en el MIT en el que se expone que incluso teléfonos inteligentes rooteados pueden ser víctimas del robo de información personal o estar expuestos a ataques de denegación de servicio.
Ali, Balushi, Nadir, & Hussain (2018)	2018	Journal	Documento que habla de las diferentes amenazas para sistemas ciber-físicos, redes de sensores, entre otros. Presenta una extensa tabla donde se exponen los ataques y sus efectos, los métodos de detección defensivos y ofensivos.
Rani, Modares, Sriram, Mikulski, & Lewis (2015)	2015	Journal	Artículo que muestra algunos ataques que se llevan a cabo en vehículos no tripulados. Se expone el riesgo que estos corren por la naturaleza de su conectividad. e muestran pruebas acerca de algunos ataques para desconexión y control de dichos dispositivos. Se explica la importancia del streaming confiable para el control y visualización de los vehículos no tripulados.

Mientras que los ‘surveys’ se encargaron de enlistar las falencias en la seguridad informática, hay información existente y pragmática que revela la peligrosidad y la diversidad que pueden afectar. Las preguntas anteriormente se responden de la siguiente manera en la tabla 1: Todos los ataques enlistados por las investigaciones recientes son aplicables a toda la diversidad de dispositivos y aplicaciones IoT. ¿Qué hay de las contramedidas? Información de expertos de ciberseguridad destacan que los ataques informáticos son cada vez más sofisticados (Cisco, 2018). También, algunos artículos indican

que dichas contramedidas no son suficientes y por eso se opta por recurrir a medidas más especializadas como el caso de los sistemas de detección de intrusiones (*IDS* por sus siglas en inglés) aplicando inteligencia artificial[6].

Durante las primeras búsquedas acerca de la amenaza en el contexto de ataques informáticos, los ataques de denegación de servicio (*DoS* por sus siglas en inglés) fueron los más abordados, presentándose casi en todos los trabajos que diseñaron y elaboraron un sistema de detección de intrusiones. Casi en su totalidad, los artículos más recientes tratan del desarrollo de algoritmos de máquinas de aprendizaje, integrando así una labor de inteligencia artificial en los nuevos trabajos. Por tal motivo, se detalló la búsqueda intentando encontrar los trabajos que tuvieran como propósito hacer detección inteligente de ataques de denegación de servicio en el internet de las cosas, así como la estrategia empleada para hacer dicha tarea. En la *Tabla 2* se muestran los artículos que sirvieron como referencia para obtener un panorama más completo de lo que se ha estado haciendo recientemente.

Tabla 2. Breve introducción de los artículos estudiados en el estado del arte.

Trabajo	Propósito	Estrategia de detección	Características del trabajo
Gajewski, Batalla, Mastorakis, & Mavromoustakis (2019)	Artículo informativo que concentra información acerca de los distintos enfoques de despliegue de los sistemas de detección de intrusiones en redes de hogares inteligentes. Propone un nuevo modelo o enfoque para hacer detección de intrusiones.	Propuesta de un modelo basado en el procesamiento de cómputo en la nube. Este procesamiento debe hacerlo el proveedor de servicios.	-Toma en cuenta los recursos limitados de los dispositivos finales para hacer IDS basado en host. -El modelo propuesto aprovecha las propiedades de hardware de equipos potentes como servidores en la nube para hacer detección de intrusiones en red.
Procopiou, Komninos, & Douligeris (2019)	Artículo que expone un sistema de detección de intrusiones basado en análisis estadístico para recolectar información del comportamiento de tráfico de red y determinar comportamientos potenciales maliciosos.	La estrategia empleada es un algoritmo de peso ligero que analiza la información de variables de red, estimando un error para cada una. Si se encuentran cambios abruptos en una determinada serie de tiempo, se encuentra tráfico potencialmente malicioso.	-Diseño basado en red. -Simulación de nodos estáticos por medio de NS-3. -Desarrollo de un algoritmo para el consumo bajo de recursos de red. -Innovación en el método de detección. -Alta tasa de detección. -Adaptado a ataques de inundamiento y denegación de servicio distribuido.
Chen, Meng, Shan, Fu, & Bhargava (2019)	Se propone el nuevo ataque LDoS al protocolo de enrutamiento para evaluar el mecanismo de seguridad y confianza en una red de sensores inalámbricos (WSN). La transformación	. Se utiliza un enfoque de análisis conjunto de frecuencia-tiempo de la transformación de Hilbert-Huang (HHT) para analizar	-Detección de ataques sigilosos como denegación de servicio de tasa baja. -Probado con componentes reales en una red Zigbee. -Alto índice de detección.

[6] (Lu & Xu, 2019), (Elrawy, Awad, & Hamed, 2018), (Yaacoub, Noura, Salman, & Chehab, 2020)

	de Hilbert-Huang y los enfoques de evaluación de confianza se combinan para detectar el nuevo ataque LDoS en zigbee WSN.	la pequeña señal no estacionaria producida por el ataque LDoS. Los enfoques de coeficiente de correlación y prueba de Kolmogorov-Smirnov (KS) se unen para evaluar la fiabilidad de los componentes del FMI y excluir los componentes falsos del FMI	
Anthi, Williams, Słowińska, Theodorakopoulos, & Burnap (2019)	Este documento propone un Sistema de detección de intrusos (IDS) de tres capas que utiliza un enfoque supervisado para detectar una variedad de ciberataques populares basados en redes en redes IoT. El sistema se evalúa dentro de un banco de pruebas de hogares inteligentes que consta de 8 dispositivos populares disponibles en el mercado. El rendimiento de las tres funciones principales del sistema da como resultado una medida F de: 1) 96.2%, 2) 90.0% y 3) 98.0%.	El sistema consta de tres funciones principales: 1) clasificar el tipo y perfil del comportamiento normal de cada dispositivo IoT conectado a la red, 2) identifica paquetes maliciosos en la red cuando ocurre un ataque, y 3) clasifica el tipo de ataque eso ha sido desplegado.	-Utilizado en el contexto de hogares inteligentes (Smart Home). -El sistema es sometido a la evaluación de 12 ataques diferentes. -Utiliza un método distribuido de detección para clasificar primeramente comportamientos anormales y luego puntualizar el ataque. -Alto índice de detección.
Jan, Ahmed, Shakhov, & Koo (2019)	Se propone utilizar el algoritmo SVM y evaluarlo al enfoque de redes neuronales, otras técnicas de machine learning y otros algoritmos	El entrenamiento del algoritmo propuesto consiste en hacer una extracción de características de variables estadísticas dadas por un conjunto de datos. El resultado provee una muestra de datos etiquetados para entrenar una modificación del clasificador SVM	-No está probado en tiempo real. -Medida F por encima del 98%. -Pruebas realizadas en un entorno simulado. -Tiempo de procesamiento hasta 200 veces menor que las técnicas SVM convencionales.
Lopez-Martin, Carro, Sanchez-Esguevillas, & Lloret (2017)	El método propuesto se basa en un autocodificador variacional condicional con una arquitectura específica que integra las etiquetas de intrusión dentro de las capas del decodificador. El método propuesto es menos	Análisis de tráfico de red. Se mapea un conjunto de parámetros para definir un primer conjunto con etiquetas de ataques y tráfico normal,	-Recuperación de variables faltantes en registros incompletos. -Alto rendimiento del modelo desarrollado: cerca de 99%. -Emulación para desarrollo del IDS y su red con

	complejo que otros métodos no supervisados basados en un codificador automático variacional y proporciona mejores resultados de clasificación que otros clasificadores conocidos.	realizado por un codificador. Un decodificador se encarga de aplicar probabilidades de las muestras para estimar ataques. Se añade un módulo de reconstrucción de características para los registros incompletos.	Tensorflow y el paquete de Python *scikit-learn.* -Imitación de una red tipo jerárquica.
Bostani & Sheikhan (2017)	En este trabajo se introducen agentes de detección de intrusiones basado en especificaciones El método propuesto determina los ataques a través de un mecanismo de votaciones. El objetivo es detectar ataques de inundamiento de paquetes, ataques de ruteo (*forwarding*) y ataques de hoyo de gusano.	Un agente o programa es instalado en cada nodo ruteador para aplicar técnicas de máquinas de aprendizaje acerca del comportamiento. Toda la información es enviada como paquetes de red a un nodo raíz que analiza cada uno de los resultados. Además, el ruteador raíz puede detectar anomalías de manera global en paralelo a los otros nodos. Para la puntualización de un comportamiento sospechoso, se utiliza un sistema basado en votaciones.	-Aplicado a un escenario de ciudades inteligentes. -El enfoque de detección es en red más basado en host (híbrido). -No se involucra la medida F para la evaluación del sistema. -Se alcanzan tasas de detección de hasta un 96%. -Pruebas realizadas con nodos físicos *Waspmote Mote Runners.* -Protocolo sobre el que se trabajó es 6LowPAN.
Raza, Wallgren, & Voigt (2013)	Hecho para redes 6LowPAN, el trabajo es uno de los primeros híbridos en sistemas de detección. En este documento se diseñó, implementó y evaluó un novedoso sistema de detección de intrusos para el IoT que llamaron SVELTE. Para su implementación y evaluación, se apunta principalmente a ataques de enrutamiento, como información falsificada o alterada, sumidero y reenvío selectivo. Sin embargo, el enfoque puede ampliarse para detectar otros ataques.	Se diseñaron 3 componentes para hacer la detección de intrusiones: mapeador, que recaba información acerca de la red RPL; Detector de intrusiones, analiza los datos provenientes desde el mapeador y estadísticamente clasifica la información; mini firewall, diseñado para filtrar tráfico no requerido y quitar carga a los nodos finales.	-Desarrollado específicamente para ataques de enrutamiento. -Utilizado en una topología jerárquica mezclada con malla. -No se utilizaron técnicas de inteligencia artificial. -Las pruebas se realizaron sobre nodos físicos.

Brun, Yin, & Gelenbe (2018)	El objetivo es detectar los ataques lanzados a las puertas de enlace de las redes del internet de las cosas. Se presentan el principio y diseño de un enfoque basado en aprendizaje de una red neuronal aleatoria. Debido a que abordan variantes poco sofisticadas de algunos ataques, se determina que la implementación de un detector simple alcanza niveles considerables de detección.	Se coloca un capturador de paquetes justo en la puerta de enlace. Este hace un análisis del flujo local, así como el que sale a internet. Luego se determinan las métricas para la clasificación, las cuales son a nivel de paquetes de red y que discriminan entre los ataques.	-Enfocado a una red jerárquica Smart home. -Se utiliza un conjunto de datos construido propiamente en un proyecto adjunto. -Las pruebas se realizaron en una red heterogénea involucrando el protocolo Zigbee, Bluetooth y WiFi.
Shone, Ngoc, Phai, & Shi (2018)	En este artículo se propone un modelo como combinación de aprendizaje profundo y superficial que es capaz de analizar un amplio rango del tráfico de red. Las técnicas combinadas son el Autocodificador profundo no simétrico (NDAE) y Random Forest (aprendizaje superficial). Las contribuciones que se ofrecen son las siguiente: -Una nueva técnica NDAE para aprendizaje supervisado, el cual provee reducción de dimensionalidad de datos no simétricos. -Un novedoso modelo clasificador que combina aprendizaje profundo con aprendizaje superficial para explotar sus respectivas fortalezas y reducir la sobrecarga de análisis.	El diseño del clasificador consiste en un modelo codificador de variables por medio de redes neuronales. Este hace la labor de extraer las características y, sin decodificar la salida, se envía al clasificador Random Forest para hacer la clasificación final. Es de mencionarse que se añadieron dos codificadores en serie antes de pasar los datos al clasificador random forest.	-Se mejora la clasificación en comparación con métodos principales como redes de creencias profundas. -Se reduce considerablemente el tiempo de entrenamiento. -Entrenamiento utilizado con los conjuntos de datos KDD Cup ’99 y NSL-KDD. -Pruebas utilizadas con Tensorflow. -Se evaluó el modelo con la medida F, así como la tasa de falsos positivos. -Se introduce el tiempo de entrenamiento como métrica de evaluación.
Deng, Li, Yao, Cox, & Wang (2018)	Se analizan las características de la seguridad de la red y los problemas de seguridad, y se discute el marco del sistema de seguridad de Internet y algunas tecnologías de seguridad clave, incluida la gestión de claves, autenticación y control de acceso, seguridad de enrutamiento,	Con redes bayesianas, se calcula la probabilidad de invasión, así como la probabilidad de anomalía. La defensa consiste en aplicar tecnologías, protocolos y medidas de seguridad probadas, para incluir	-No se hace uso de la medida F para la evaluación del modelo. -La topología utilizada es jerárquica. -No se especifica la cama de pruebas. -Se utiliza el procesamiento de los datos basado en la nube. -Se remarcan los verdaderos positivos y falsos positivos en sus pruebas.

	protección de la privacidad, detección de intrusos y tolerancia a fallas e intrusiones.	el algoritmo desarrollado. El procesamiento se realiza en un enfoque basado en la nube, aplicando técnicas de minería de datos.	
Diro & Chilamkurti (2018)	Artículo que propone hacer detección de ataques a través de un sistema distribuido llamado cómputo en la nube. En su artículo se pretende demostrar que las técnicas de aprendizaje profundo son mejores que las de aprendizaje superficial en cuanto a precisión.	La técnica de detección está basada en cómputo en la niebla. Cada nodo en la niebla maneja el mismo modelo de detección, el cual es actualizado por un nodo maestro cuando hay cambios. Se comunican entre sí para hacer sinergia entre el entrenamiento y las pruebas.	-Se enfocan a ciudades inteligentes. -La topología de red es jerárquica. -Tiempo de respuesta efectivo. -Uso de procesamiento distribuido. -Aumento en el tiempo de entrenamiento del modelo de aprendizaje profundo. -Se indica una figura con el tiempo de detección sin indicar lo que significa.
Justin, Marathe, & Dongre (2017)	Trabajo de dominio teórico que propone un sistema de detección de intrusiones con aplicación híbrida en cuanto a las reglas: basada en firmas y basada en anomalías. También se consideraría un IDS híbrido por la distribución de los nodos: agentes en cada nodo y uno maestro. Se advierte sobre el ataque de denegación de servicio como problema y se presenta un diagrama de actividad de los nodos en el que todo el tráfico pasa por un nodo en específico.	A cada nodo se le incluye un modelo de detección basado en SVM para hacer una detección propia de anomalías. Además, el agente incluye firmas, de esta manera, al fallar la detección basada en firmas, el algoritmo SVM comienza a hacer la clasificación. Se añade una fase colaborativa donde los nodos pueden decidir acerca del comportamiento anómala para un solo nodo.	-Hecho para una red inalámbrica multisalto. -Sistema híbrido de detección. -Uso de KDD como conjunto de datos. -Propuesta de trabajo. -No incluye fase de pruebas y resultados.
Zardari, y otros (2019)	En este artículo se propone una técnica llamada detección de ataque dual para ataques de hoyo negro y gris, realizada para redes adhoc móviles. Además de tratar dichos ataques como variantes de denegación de servicio, se explica la lógica de ataque de cada uno.	La metodología de detección consta de identificar a los nodos maliciosos que envían paquetes de secuencia con valores altos o constantes para asegurar que son la mejor ruta. Posteriormente, se verifica si estos nodos están eliminando los paquetes recibidos y	-Probado en un ambiente simulado con Network Simulator 2. -Aborda ataques de hoyo negro y gris como ataques DoS. -AODV como protocolo de enrutamiento. -Se simula el protocolo Zigbee para redes MANET. -Rendimiento en el umbral de 95-100.

		las razones de ello (por ejemplo, modificando el valor *queue* de las tramas).	
Elrawy, Awad, & Hamed (2018)	Este artículo presenta una encuesta exhaustiva de los últimos IDS diseñados para el modelo de IoT, con un enfoque en los métodos, características y mecanismos correspondientes.	N/A	Análisis de IoT desde su arquitectura genérica de 3 capas. Argumentación de la importancia de hacer uso de cómputo en la nube para procesamiento de los datos. Asegura que deben protegerse 3 aspectos en una red IoT: Disponibilidad, Integridad y Confidencialidad. También explica los dos tipos de IDS: basado en host y en red. Denota la importancia de hacer reducción de características al hacer uso de máquinas de aprendizaje
Tweneboah-Koduah, Skouby, & Tadayoni (2017)	Desde la perspectiva de las configuraciones de infraestructura de firmware, hardware y software, este documento analiza algunos de los principales dominios de aplicaciones y servicios de IoT, y analiza los desafíos de ciberseguridad que probablemente impulsen la investigación de IoT en el futuro cercano.	N/A	Muestra un estudio del 2015 que indica a los ataques DoS y de ejecución de código como los más comunes. Presenta un caso de uso junto con scripts para hacer ataques DoS e inyección SQL. Demuestra la vulnerabilidad verdadera por cualquier ataque a IoT.
Pecorella, Pierucci, & Nizzi (2018)	El documento sugiere adaptar dinámicamente el nivel de seguridad de la red doméstica inteligente de acuerdo con el nivel de riesgo percibido por el usuario, lo que se denomina como análisis de sentimiento de red. La seguridad de la red doméstica inteligente se mejora mediante firewalls distribuidos y sistemas de detección de intrusiones tanto en el lado del hogar inteligente como en el lado del proveedor de servicios de Internet.	Plantea una arquitectura de 3 entidades conectadas entre sí que consta de un SH (Shield home), SLU (Shield Logic Unit) y SN (SCN). El SN es la modificación del firewall directamente instalado en el proveedor de servicios, el segundo se considera el núcleo de la arquitectura, ya que recibe todas las alarmas y advertencias. El primero es para armonizar todas las conexiones.	-Aplicación a entorno SmartHome. -Uso de topología jerárquica. -Modificación dinámica de los parámetros de seguridad. -Uso de nodos físicos para la realización de pruebas. -Uso de los protocolos Ethernet, WiFi y 802.15.4. -Centrado en la prevención y no en la detección.

Jokar & Leung (2018)	En este artículo se presenta un novedoso sistema de detección y prevención de intrusos para redes de área doméstica basadas en Zigbee en redes inteligentes, HANIDPS. HANIDPS emplea un mecanismo de detección de intrusos basado en modelos, así como un sistema de prevención de intrusos basado en aprendizaje automático para proteger la red contra una amplia gama de tipos de ataques	Modelo que emplea un sistema de detección y un sistema de prevención de intrusiones. Captura datos directamente de la red, extrae las características más relevantes y estima el estado del mismo. La técnica de identificación es por medio de Q-learning.	-Uso de Q-learning. -Probado en un entorno simulado. -Aplicación real de reglas para la gestión de paquetes por medio del sistema de prevención. -La tasa de detección es la única métrica empleada.
Takase, Kobayashi, Kato, & Ohmura (2019)	Se propone un mecanismo de detección de malware utilizando valores extraídos del procesador. El objetivo es descargar el mecanismo de detección de malware al hardware mediante el uso de la información del procesador y se pretende suprimir el consumo de recursos de hardware causado por ataques DoS.	Mecanismo físico insertado en la placa del componente o nodo para recabar información de 17 características acerca del procesador.	-No presenta una topología explícita. -Variables tomadas directamente del microprocesador. -Hecho para detectar malware y algunas afectaciones por DoS. -Aplicado a redes de sensores.
Alheeti & McDonald-Maier (2018)	En este artículo se propone un mecanismo de protección inteligente el cual fue creado para asegurar las comunicaciones externas para autos autónomos y semiautónomos. Dicho sistema de detección híbrido utiliza redes neuronales de retropropagación para detectar denegación de servicio (DoS).	Al no aplicarse a un entorno real, se toma como referencia el algoritmo desarrollado que consta de 4 fases: pre-procesamiento, selección de características, difusificación (fuzzification) y entrenamiento y prueba.	-Realizado para redes adhoc de vehículos. -Uso de algoritmos de lógica difusa y redes neuronales de retropropagación. -No realizado en un entorno real. -Uso de datasets KDD cup ’99 y Kyoto Benchmark. -Datasets más actualizados y gira en entornos IoT.
Condomines, Zhang, & Larrieu (2019)	Se propone un sistema de detección de intrusiones híbrido basado en el tráfico espectral para estimar anomalías en la red. Para ello se considera utilizar firmas intercambiadas en la red y apunta a las redes adhoc de vehículos aéreos no tripulados.	Se realiza una representación del sistema en una ventana de tiempo, utilizando medidas sinodales. Las variables surgen de medidas de conexión.	-Desarrollado en el contexto de FANETs. -No realiza pruebas comparativas de rendimiento. -El aporte consta de aseverar que es posible hacer IDS con representaciones espectrales del comportamiento de red de los dispositivos. -Se da detalle la cama de pruebas utilizada para armar una red FANET virtualizada.

Yang, Moubayed, Hamieh, & Shami (2019)	En este artículo, se propone un sistema inteligente de detección de intrusos (IDS) basado en modelos de aprendizaje automático de estructura de árbol. La finalidad es detectar diferentes ciberataques a un bajo costo computacional y con una tasa alta de detección.	Consta de una etapa de pre-procesamiento de las características, otra de selección de las mismas y posteriormente se utilizan técnicas de máquinas de aprendizaje basados en árboles de decisiones y árboles aleatorios.	-Bajo costo computacional. -Hecho para VANETs. -Probado con los datasets CAN-Intrusion y CICIDS2017. -Baja tasa de falsos positivos y alta evaluación de F1.
Pereira, y otros (2020)	Artículo que propone un protocolo de comunicación basado en Zigbee para vehículos no tripulados. A través de una comparación contra LTE y WiFi, se pretende mostrar a Zigbee como una opción que, entre otras ventajas, no cuenta con una infraestructura propia.	N/A	-Implementación de una VANET empleando Zigbee. -Los módulos empleados son XBee Pro S3B. -Se utiliza Raspberry Pi3 como placa asociada al módulo de comunicación. -Rendimiento de entrega exitosa de paquetes streaming igual que WiFi y LTE.
Yaacoub, Noura, Salman, & Chehab (2020)	El objetivo de este trabajo es investigar las amenazas emergentes del uso de drones en ciberataques, junto con las contramedidas para frustrar estos ataques. También se revisan los diferentes usos de los drones con fines maliciosos, junto con los posibles métodos de detección. Como tal, este documento analiza la explotación de las vulnerabilidades de los drones dentro de los enlaces de comunicación, así como dispositivos y hardware inteligentes, incluidos teléfonos inteligentes y tabletas.	N/A	-Se exponen las diferentes aplicaciones de los vehículos no tripulados. -Se muestran las diferentes técnicas de ataques cibernéticos que las redes de vehículos pueden sufrir. -Se consideran las redes adhoc de vehículos terrestres (VANET) y voladores (FANET). -Se detallan algunos requerimientos operacionales de algunas regiones. -En cuanto a los ciberataques se describe en detalle su naturaleza, el requerimiento de seguridad afectado y algunas contramedidas.

La investigación arrojó más artículos científicos que abordaban los ataques de denegación de servicio, sin embargo, la tabla 2 solo muestra los más importantes y de mayor factor de impacto.

De lo obtenido, primero se comenzó a indagar en el contexto de redes para hogares inteligentes y le siguieron las redes de sensores inalámbricos para industria 4.0, IoT médico y aplicaciones en redes pequeñas de la industria. Cuando la investigación se comenzó a encaminar hacia el protocolo Zigbee, por la naturaleza de su conectividad y su topología, se

encontró el contexto de redes móviles adhoc (MANET por sus siglas en inglés) y dos de sus especializaciones más importantes, redes adhoc de vehículos (VANET por sus siglas en inglés) y redes adhoc de vehículos voladores (FANET por sus siglas en inglés).

La tabla *Tabla 3* detalla por cada trabajo el método de detección empleado, el protocolo de comunicación, los ataques abordados, así como las variables utilizadas para su detección y por último la topología.

Tabla 3. Comparativa de las técnicas, ataques y variables abordados en el estado del arte

	Técnica	Protocolo de comunicación	Ataques abordados	Variables	Topología de red
Gajewski, Batalla, Mastorakis, & Mavromoustakis (2019)	Sí, aunque no especifica	WSN	N/E	N/A	Jerárquica
Procopiou, Komninos, & Douligeris (2019)	Forecasting + Teoría del Caos	Emulación WSN	-Ataques de inundación DoS -Denegación de servicio de tasa baja (LDoS)	-Número de consultas -Número de paquetes -Tasa de datos -Tamaño promedio de los paquetes -Tiempo promedio entre respuestas -Tiempo promedio entre cada consulta y respuesta -Tiempo promedio entre consultas -Consultas paralelas	Jerárquica
Chen, Meng, Shan, Fu, & Bhargava (2019)	Transformada de Hilbert Huang + evaluación de confiabilidad	zigbee + WSN	Denegación de servicio de tasa baja (LDoS)	-RREQ number -Frecuencia de fase -Tiempo (s)	Jerárquica
Anthi, Williams, Słowińska, Theodorakopoulos, & Burnap (2019)	Naive Bayes, Redes bayesianas, J48, Zero R, One Zero, Logística simple, SVM, Preceptrón multicapa, RF	WiFi + Zigbee	-Ataques de escaneo. -DoS/DDoS. -MITM. -Replay. +Envenenamiento ARP y DNS	35 variables con valores de conexiones TCP e ICMP.	Jerárquica

Jan, Ahmed, Shakhov, & Koo (2019)	SVM propuesto, GA-SVM, A-IDS, WPS-IDS	Emulación	-Ataques DDoS	78 variables con valores estadísticos	Jerárquica
Lopez-Martin, Carro, Sanchez-Esguevillas, & Lloret (2017)	Random Forest, Linear SVM, Multinomial	Emulación	-DoS -Probe -R2L -U2R	41 características del dataset que se convierten en 116 por la transformación de datos	Jerárquica
Bostani & Sheikhan (2017)	Optimum-Path Forest, Clustering, SA-IDSs	Emulación RPL en 6LowPAN	-Ataque de hoyo de gusano -Sinkhole attack -Rank attacks	-Tasa de paquetes recibidos -Tasa de paquetes descartados. -Latencia promedio. -Contador máximo de saltos.	Jerárquica
Raza, Wallgren, & Voigt (2013)	N/A	RPL	-Sinkhole -Selective forwarding	-Rango del nodo hijo. -Rango del nodo padre. -Fallas del nodo. -Límite de fallas.	Jerárquica
Brun, Yin, & Gelenbe (2018)	Random Neural Networks	RF + WiFi + ZB + Bluetooth	-TCP SYN Attacks. -Inundamiento UDP. -Ataque de privación del sueño. -Ataques broadcast.	Número de paquetes salientes de ICMP "destino inalcanzable". -Número de conexiones TCP medio abiertas. -Paquetes enviados sobre una escala de tiempo. -Número de paquetes broadcast.	Jerárquica
Shone, Ngoc, Phai, & Shi (2018)	Non-symmetric Deep auto-encoder + Random Forest	Dataset	-DoS -Probe -R2L -U2R	41 características presentadas en los datasets aplicando reducción de características. No especifican el resultado	Jerárquica
Deng, Li, Yao, Cox, & Wang (2018)	Rough Set + SVM + Principal component analysis	Dataset	-DoS -U2R -R2L -Probing	42 características de los datasets que se reducen con ayuda del algoritmo PCA. No especifican el resultado.	Jerárquica
Diro & Chilamkurti (2018)	Aprendizaje profundo	Emulación de cómputo en la niebla	-DoS -Probe -R2LU2R	41 características de los datasets.	Jerárquica

	(Keras on Theano)				
Justin, Marathe, & Dongre (2017)	Basado en SVM + firmas	N/A	Denegación de servicio	N/E	Multisalto – Jerárquico
Zardari, y otros (2019)	N/E	Simulación Zigbee	-Ataque de hoyo negro -Ataque de hoyo gris	-Estatus de los paquetes: Números de secuencia, reconocimiento, entrega y tasa de eliminación de paquetes. -Estatus del nodo: Batería, intensidad de señal, carga de red.	Distribuido
Pecorella, Pierucci, & Nizzi (2018)	Basado en firmas	WiFi + Zigbee	-Ransomware -Escaneo de puertos -Ataques de enrutamiento	N/E	Distribuido
Jokar & Leung (2018)	Q-learning	Zigbee	-Interferencia de radio -Protección de reenvío -Esteganografía -Manipulación Back-off -Denegación de servicio	-Datagramas -Tasa de tráfico -RSS -Número de secuencia -Tasa de error de paquetes -Disponibilidad de los nodos	Jerárquico
Takase, Kobayashi, Kato, & Ohmura (2019)	Análisis de anomalías modificando el hardware del dispositivo final	N/E	Malware	17 características pertenecientes exclusivamente al microprocesador	N/E
Alheeti & McDonald-Maier (2018)	Identificación de anomalías con Perceptron Multicapa	Zigbee (Por la naturaleza de la aplicación)	-DoS	Características extraídas de Kyoto-dataset	Distribuido
Condomines, Zhang, & Larrieu (2019)	Análisis espectral basado en firmas con un análisis multi-fractal Wavelet Leader	Zigbee para FANET's	-DDoS	-Congestión TCP -QUEUE *No especifica el resto de las variables	Distribuido

Yang, Moubayed, Hamieh, & Shami (2019)	Árboles de decisiones + Random Forest + Árboles extras + XGBoost	N/A	-DoS -Escaneo de puertos -Fuerza bruta	Variables concernientes a Car-Hacking dataset	Distribuido

Los trabajos en mejores revistas casi siempre mostraron algoritmos, técnicas, modelos o soluciones empleando máquinas de aprendizaje. Para encontrar el nivel de progreso y evaluación de cada uno, se desarrolló la *Tabla 4* en donde se muestran las métricas más empleadas de entre los artículos (Tasa de detección y medida F) así como donde más hubo falencias o en su defecto fueron faltantes (Tasa de falsos positivos y Tiempo de detección de cada ataque realizado).

Tabla 4. Tabla de rendimiento de los algoritmos propuestos en el estado del arte.

	Tasa de detección	F-Score	FPR	Tiempo de detección
Gajewski, Batalla, Mastorakis, & Mavromoustakis (2019)	N/A	N/A	N/A	N/A
Procopiou, Komninos, & Douligeris (2019)	66.67% - 100%	N/E	0.75% - 100%	N/A
Chen, Meng, Shan, Fu, & Bhargava (2019)	N/E	N/E	N/E	N/A
Anthi, Williams, Słowińska, Theodorakopoulos, & Burnap (2019)	80% - 98.8%	88.8% - 99.0%	0.05% - 66%	N/A
Jan, Ahmed, Shakhov, & Koo (2019)	89.76% – 98.03%	N/E	6% - 73%	N/A
Lopez-Martin, Carro, Sanchez-Esguevillas, & Lloret (2017)	65% - 99%	88.39%	1.12% - 15.59%	N/A
Bostani & Sheikhan (2017)	96.02%	N/E	8.85%	N/A
Raza, Wallgren, & Voigt (2013)	80% - 100%	N/E	N/E	N/A
Brun, Yin, & Gelenbe (2018)	N/E	N/E	N/E	N/A

Shone, Ngoc, Phai, & Shi (2018)	N/E	97.85%	50.00%	N/A
Deng, Li, Yao, Cox, & Wang (2018)	96.8%	N/E	1.6%	N/A
Diro & Chilamkurti (2018)	71% – 99.52%	95.65% - 99.14%	0.85% - 6.57%	N/A
Justin, Marathe, & Dongre (2017)	N/A	N/A	N/A	N/A
Zardari, y otros (2019)	98.15%	N/A	N/A	N/A
Pecorella, Pierucci, & Nizzi (2018)	N/A	N/A	N/A	N/A
Jokar & Leung (2018)	N/E	N/E	N/E	N/A
Takase, Kobayashi, Kato, & Ohmura (2019)	100%	N/A	N/E	N/A
Alheeti & McDonald-Maier (2018)	99.23%	N/A	1.65% - 2%	N/A
Condomines, Zhang, & Larrieu (2019)	N/A	N/A	N/A	N/A
Yang, Moubayed, Hamieh, & Shami (2019)	99.8%	99.8%	0.011% - 5.6%	N/A

La *Tabla 5* muestra una comparación de los materiales y el entorno utilizado para realizar las pruebas en cada trabajo. Los trabajos que emplearon conjuntos de datos establecidos por terceros son aquellos que hicieron un algoritmo de máquinas de aprendizaje centrándose en proponer un método diferente de detección. A diferencia de los anteriores, hubo trabajos que trabajaron de la misma manera dándole más importancia al despliegue en tiempo real y en entornos simulados o reales, utilizando materiales especializados para una conectividad verdadera en IoT. Dado que la realización de pruebas es un factor determinante al momento de publicar revistas JCR, se investigó el ambiente, las herramientas o materiales, el conjunto de datos, la metodología de pruebas utilizada, los nodos maliciosos y benignos que validaran una red creada.

Tabla 5. Cama de pruebas de los trabajos estudiados.

Trabajo	Ambiente	Herramienta/Materiales	Conjunto de datos	Metodología de pruebas utilizada	Nodos legítimos	Nodos maliciosos
Gajewski, Batalla, Mastorakis, & Mavromoustakis (2019)	N/A	N/A	N/A	N/A	N/A	N/A
Procopiou, Komninos, & Douligeris (2019)	Simulación	Network Simulator 3	Propio	N/E	7	2
Chen, Meng, Shan, Fu, & Bhargava (2019)	Real	Nodos Zigbee CC2530 SoC	Propio	N/E	6	3
Anthi, Williams, Słowińska, Theodorakopoulos, & Burnap (2019)	Real	Lifx Smart lamp Hive Hub 6 dispositivos diferentes más que no utilizan Zigbee	Propio	N/E	8	1
Jan, Ahmed, Shakhov, & Koo (2019)	Simulación	Matlab 2018b	Propio	N/E	N/E	N/E
Lopez-Martin, Carro, Sanchez-Esguevillas, & Lloret (2017)	N/A	N/E	NSL-KDD	N/E	N/A	N/A
Bostani & Sheikhan (2017)	Simulación	Matlab R2014a	Propio	N/E	10	6
Raza, Wallgren, & Voigt (2013)	Simulación	Contiki OS Cooja network simulator	Propio	N/E	32	4
Brun, Yin, & Gelenbe (2018)	Real	Sensores Zigbee Sensores RF869 Sensores Bluetooth	Propio	N/E	8	1
Shone, Ngoc, Phai, & Shi (2018)	Simulación	TensorFlow simulator	KDD Cup ‘99	N/E	N/A	N/A

			NSL-KDD			
Deng, Li, Yao, Cox, & Wang (2018)	Simulación	Matlab	KDD Cup ‘99	N/E	N/A	N/A
Diro & Chilamkurti (2018)	N/E	N/E	NSL-KDD	N/E	N/A	N/A
Justin, Marathe, & Dongre (2017)	N/A	N/A	N/A	N/A	N/A	N/A
Zardari, y otros (2019)	Simulación	Network Simulator 3	N/E	N/E	100	N/E
Pecorella, Pierucci, & Nizzi (2018)	Simulación / Real	OpenMote CC2538 UDOO Board	N/E	N/E	N/E	N/E
Jokar & Leung (2018)	N/E	N/E	N/E	N/E	N/E	N/E
Takase, Kobayashi, Kato, & Ohmura (2019)	Simulación / Emulación	CBP Emulator QEMU	Propio	N/E	N/E	N/E
Alheeti & McDonald-Maier (2018)	N/E	N/E	KDD Cup ‘99 Kyoto dataset	N/E	N/E	N/E
Condomines, Zhang, & Larrieu (2019)	Simulación	OMNET++ Máquinas virtuales Paparazzi Software	N/E	N/E	N/E	N/E
Yang, Moubayed, Hamieh, & Shami (2019)	Emulación	Python 3.5 + computadora core i7 8th	CAN-intrusion dataset	N/E	N/E	N/E

CAPÍTULO 4. METODOLOGÍA

En este capítulo se describe la metodología utilizada para simular una red de vehiculos no tripulados con redes zigbee. Además, se describe cómo se han replicado los ataques y el flujo para aplicar la detección de intrusiones.

4.1 Arquitectura de 3 fases

En la arquitectura de Yang, Moubayed, Hamieh, & Shami (2019) se describe un modelo donde se hace un caracterización de los dispositivos de red en una primer fase, determinando anomalías en una segunda fase.

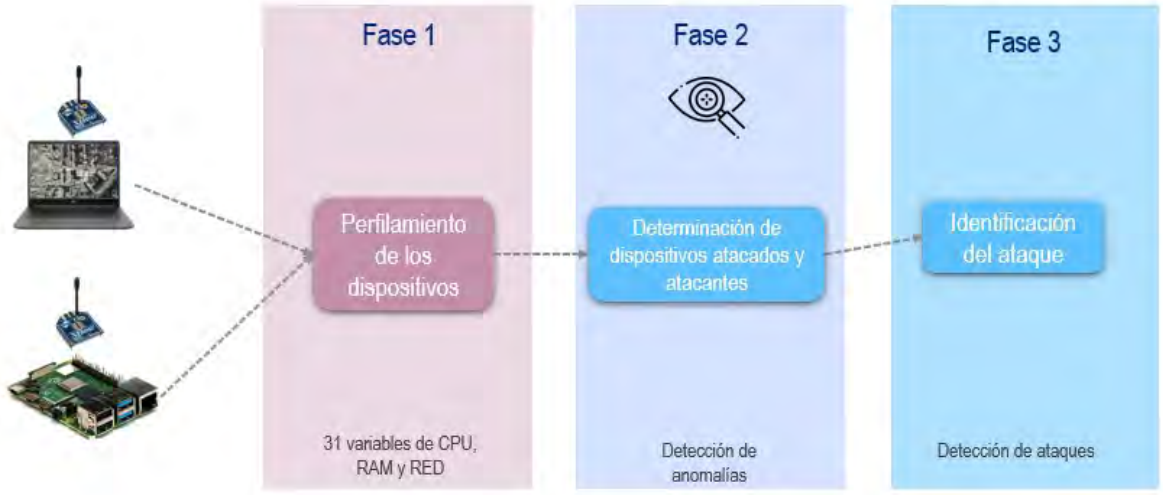


Fig. 5. Esquema de detección basado en Yang, Moubayed, Hamieh, & Shami (2019).

En la figura Fig. 5 se propone una arquitectura de 3 fases basada en la arquitectura propuesta en Yang, Moubayed, Hamieh, & Shami (2019) y se explica a continuación.

- **Fase 1**: El perfilamiento de los dispositivos a salvaguardar recolecta 33 variables de red, memoria RAM y CPU. En esta fase se genera, a partir de un conjunto de datos en general, subconjuntos de datos dependiendo de cada dispositivo capturado. En esta fase se genera un modelo SVM para cada dispositivo y abre el paso a avanzar a las fases 2 y 3.
- **Fase 2**: Una vez generados los modelos correspondientes, se efectúa una determinación en tiempo real de los dispositivos atacados y los atacantes. Si una anomalía es detectada, se avanza a la siguiente fase.
- **Fase 3**: En ésta última fase se determina el ataque DoS o la variante que está siendo ejecutada en un determinado dispositivo.

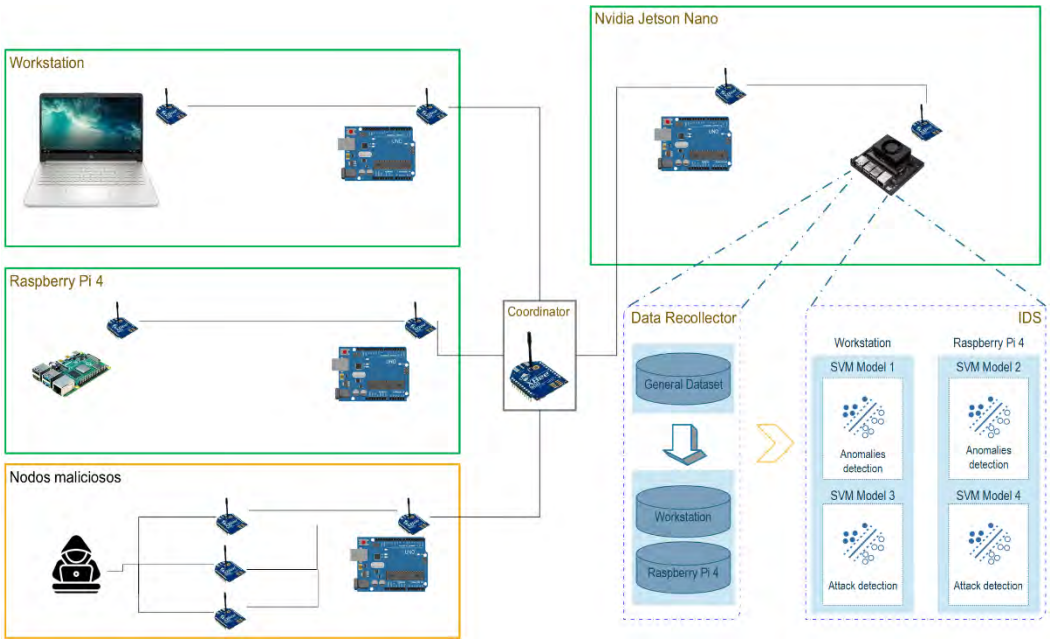


Fig. 6. Arquitectura del modelo de detección propuesto.

En el esquema planteado en la figura Fig. 6 se muestran los actores involucrados en la red zigbee. En este modelo se denota la generación de un conjunto de datos general y que contiene la información de las 33 variables recolectadas en cada uno de los dispositivos. El modelo pretende describir que en el nodo recolector de datos se concentran los conjuntos de datos y, a la vez, los modelos para los dispositivos en la red zigbee que se desean salvaguardar. En caso de tener *n* dispositivos a salvaguardar, el IDS tendría *n* modelos configurados, de los cuales serían dos por cada dispositivo.

Los nodos involucrados son los siguientes:

- Workstation. Dispositivo final que simula la recepción de paquetes de monitoreo de una tarjeta Raspberry pi 4.
- Raspberry pi 4. Dispositivo final que envía y recibe paquetes de una Workstation y que simula un vehículo no tripulado por sus características de hardware.
- Coordinador. Dispositivo central zigbee indispensable para que todos los nodos puedan conectarse a la red.
- Nodos maliciosos. Dispositivos finales que tienen la capacidad de atacar al dispositivo Raspberry pi 4.

4.2 Simulación de operación

La simulación de operaciones apunta a simular la operación entre un vehículo no tripulado que comparte imágenes en tiempo real mientras que un dispositivo final monitorea o visualiza dichas imágenes. Un dispositivo Raspberry Pi 4 y una Workstation simulan dicho proceso. Mientras éstos dos dispositivos hacen un envío/recepción de datos, de manera paralela, se envían datos críticos acerca de la carga de memoria RAM, CPU y red zigbee.

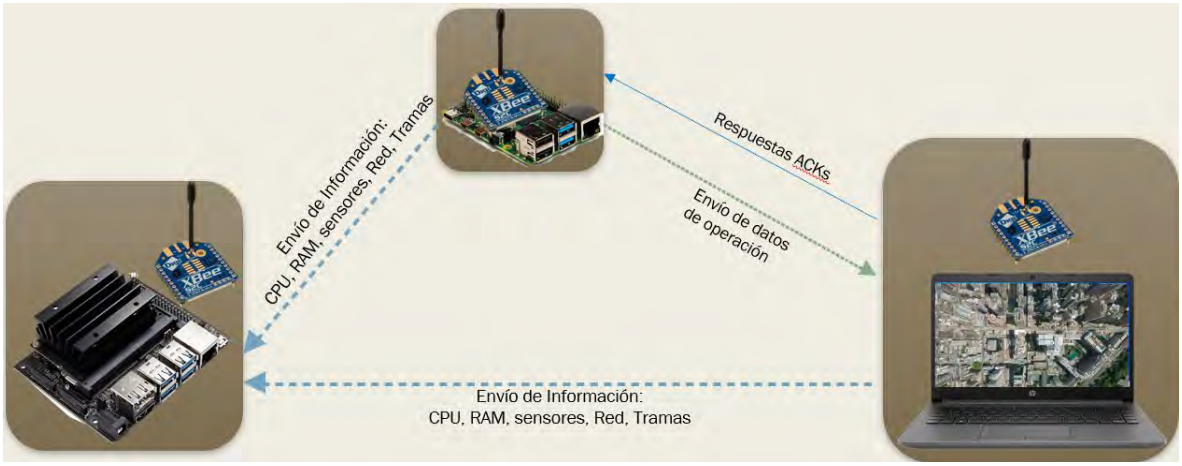


Diagrama 1. Envío y recepción de datos de dispositivos finales en la red zigbee.

El Diagrama 1 ilustra los datos que se comparten entre los dispositivos anteriormente mencionados y resalta las variables que se comparten.

4.3 Generación de Variables y conjuntos de datos

Dado que se pretende hacer una identificación de intrusiones desde dispositivos finales, en este trabajo se plantea tratar con variables que involucra una carga de red, memoria RAM y CPU. Dichas variables se generan a través de *psutils* con Python y son compartidas por medio de las tramas zigbee.

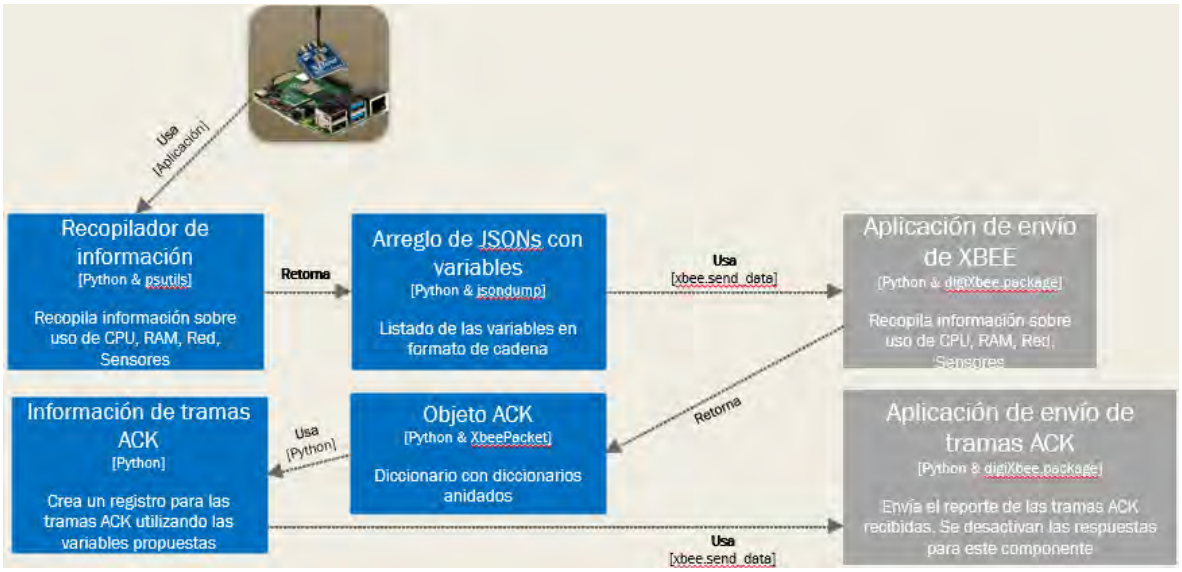


Diagrama 2. Diagrama de contenedores para el envío de información de variables a través de las tramas zigbee.

El Diagrama 2 muestra el proceso que se lleva a cabo por el agente desarrollado para compartir la información de las variables a través de las tramas zigbee. Este proceso, y agente desarrollado, está presente en cada uno de los dispositivos finales en la red.

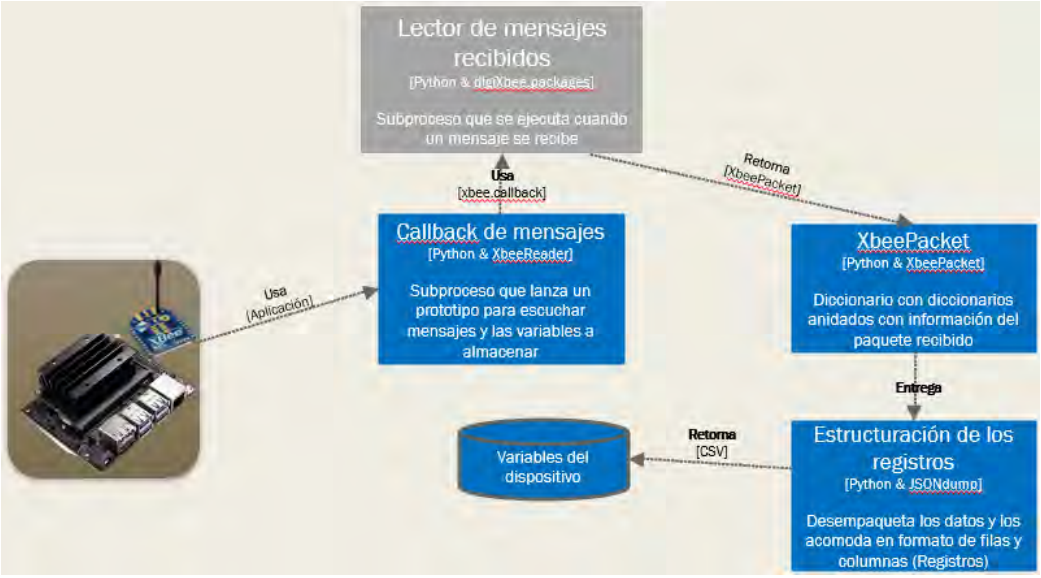


Diagrama 3. Diagrama de contenedores para la generación del conjunto de datos general.

La información generada por los dispositivos finales se transmite hacia el nodo recolector, el cual retiene la información a través de una función '*callback*'. Una vez que esta información se recibe, comienza un proceso donde se acomodan a forma de registros y columnas, escribiendo en tiempo real en un archivo de extensión *.csv*. De esta manera, el conjunto de datos general es generado (Véase Diagrama 3).

En total se utilizan más de 30 variables y son las que se listan a continuación.

- Variables de CPU: *Ctx_switches, Interrupts, Soft_interrupts, Syscall, Frec_Actual, Times_user, System, Idle, Nice, Guest_nice, Iowait, Irq, SoftIrq, Steal.*
- Variables de RAM: *Virtual_available, Virtual_used, Virtual_free, Virtual_active, Virtual_inactive, Virtual_buffers, Virtual_cached, Virtual_shared, Virtual_slab.*
- Variables de red: *entrada y salida de bytes, entrada y salida de paquetes, errores en paquetes entrantes y salientes, paquetes ACK, tipo de paquete, checksum, longitud.*

4.4 Metodología de pruebas

En este apartado se han descrito dos etapas o diseños que se implementaron en la fase de experimentación. La primera etapa sirve para encontrar el nivel de rendimiento de las variables que se proponen, haciendo una comparación con un conjunto de datos conocido. Mientras tanto, la segunda etapa propone un enfoque para la utilización de estas variables y se especifica una metodología que permite definir un modelo de SVM para cada dispositivo en la red.

4.4.1. Etapa I.

Para la etapa I, se pretende analizar las variables para un conjunto de datos existente en el internet de las cosas que haya utilizado redes zigbee en una topología distribuida. Para esto, se ha decidido utilizar el conjunto de datos *Kyoto Benchmark Data* (*KBD* por sus siglas en inglés), ya que recolecta información de vehículos aéreos no tripulados conectados a una

red zigbee distribuida a través de puntos de recolección llamados *honeypots*. Este conjunto de datos está dividido en 30 partes y cada parte se integra de 24 variables, de las cuales 14 son numéricas. Cuenta con más de 100 mil registros por parte. Es de mencionarse que para estas pruebas solo se experimentaron con 5 partes diferentes y se tomó la mitad de cada conjunto de datos.

Cabe mencionar que para estas pruebas solo se experimentaron con 5 partes diferentes y se tomó la mitad de cada conjunto de datos. Este conjunto de datos es comparado con el nuestro en donde se plantea detectar anomalías en ambos casos.

El sistema de detección de anomalías se llevó a cabo con SVM por las características de cada variable (datos numéricos). En esta primera etapa, se ha trabajado con un primer conjunto de datos propio recolectado de la arquitectura propuesta. Dicho conjunto de datos cuenta con más de 30 variables, 2 clases y más de 14 mil registros.

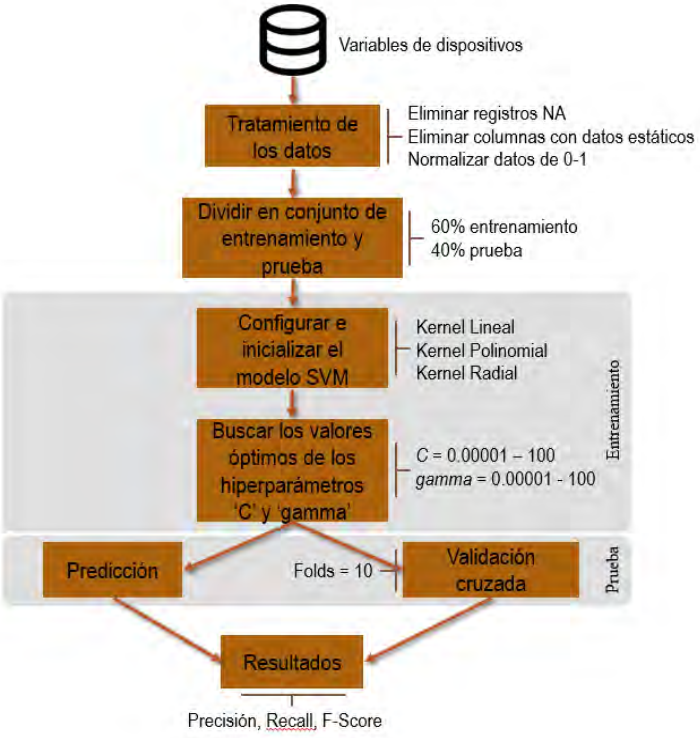


Fig. 7. Flujo utilizado para las pruebas con ambos conjuntos de datos.

A partir de los conjuntos de datos seleccionados, se da un tratamiento previo a los datos y se dividen para su respectiva fase, donde el 60% corresponde a entrenamiento y 40% para pruebas, teniendo únicamente dos categorías: benigno y anómalo. En cuanto al entrenamiento con SVM, se evalúan tres *kernels* (*Linear*, *Poli* y *RBF*) y se buscan los mejores hiperparámetros *C* y *gamma* dentro de un grid para obtener aquellos con mejores resultados, que toman valores desde 0.00001 hasta 100. La Fig. 7 muestra con más detalle lo que cada paso involucró.

4.4.2. Etapa II.

El conjunto de datos inicial contiene información de 6 distintos dispositivos en la red y hay más de 84 mil registros en total. Cuando se hace la división por registro, quedan en promedio con 14 mil registros cada uno.

Los subconjuntos de datos son utilizados para entrenar cada modelo SVM para identificar el estatus y la clase a la que corresponden, es decir, las pruebas están hechas en dos partes. La primera trata de encontrar el estatus del dispositivo a través de los modelos entrenados, mientras que la segunda parte define los ataques.

En cada una de las pruebas realizadas se definen un resultado a través de las siguientes métricas:

- Precisión: Es el porcentaje de los elementos que verdaderamente fueron clasificados correctamente. La fórmula es la siguiente:

$$\frac{TP + TN}{TP + FP + FN + TN}$$

- Sensibilidad (Recall en inglés): Es la tasa de identificación que indica lo que fue correctamente clasificado en cuanto a los verdaderos casos positivos. La fórmula de la sensibilidad es la siguiente:

$$\frac{TP}{TP + FN}$$

- Especificidad: Es la tasa de identificación que indica lo que fue correctamente clasificado en cuanto a los verdaderos casos negativos. La fórmula de la especificidad es la siguiente:

$$\frac{TN}{TN + FP}$$

- F1-Score: Resume la sensibilidad y la especificidad en una sola métrica.

$$\frac{2 \cdot \mathrm{Precision} \cdot Sensibilidad}{\mathrm{Precision} + Sensibilidad}$$

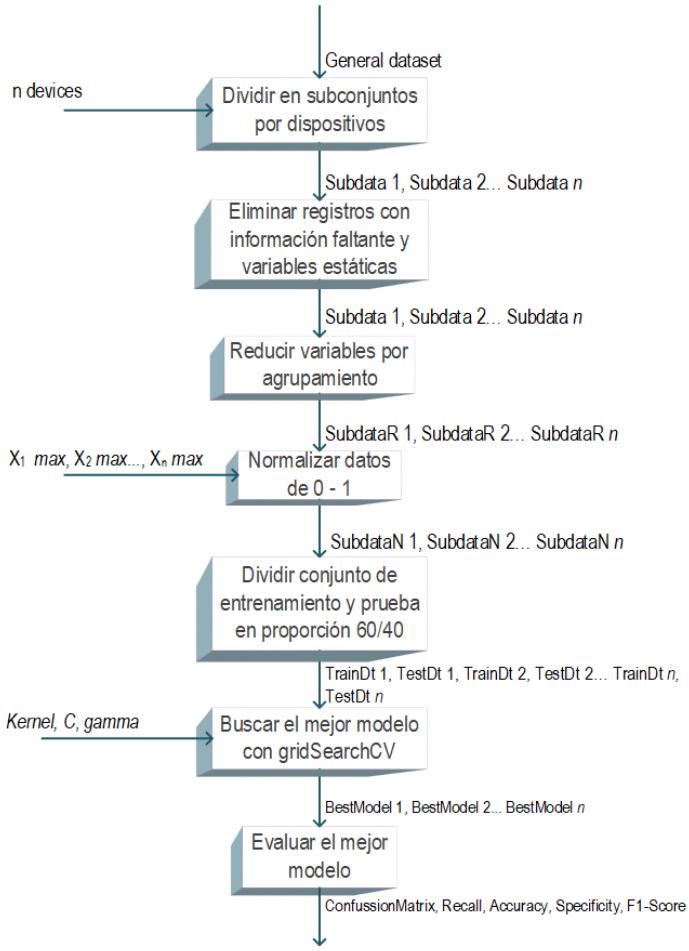


Fig. 8. Diagrama de bloques para el entrenamiento y prueba de los modelos SVM.

En la Fig. 8, se muestra la metodología utilizada para la realización de las pruebas. En ella también se considera la realización de un preprocesamiento de datos para el entrenamiento de los modelos SVM. En esta acción se involucra eliminar los registros con información faltante y también información que tenga una varianza de cero (variables estáticas).

Cabe mencionar que un 40% de los subconjuntos de datos se han destinado para el entrenamiento de los modelos mientras que el 60% restante está destinado para las pruebas.

CAPÍTULO 5. RESULTADOS

Como se ha mencionado en el Capítulo 4. , esta sección se ha divido en dos partes.

La primera parte tiene como objeto mostrar el rendimiento SVM en un conjunto de datos propio y el de *KBD*. Además, dado que se proponen nuevas variables, se pretende una identificación de anomalías con SVM a través de éstas.

La segunda etapa tiene como objetivo plantear una identificación de estados y se agrega una nueva variante de DoS para agregar complejidad al proyecto.

5.1 Etapa I.

La primera etapa de la experimentación hizo una comparativa entre el conjunto de datos *KBD* y un primer conjunto de datos propio. En primera instancia se presenta una distribución de los datos y un análisis de correlación para el primer conjunto de datos. Posteriormente, se hicieron pruebas con SVM y se exponen algunos hallazgos en la realización de estas pruebas.

5.1.1. Distribución y correlación de datos

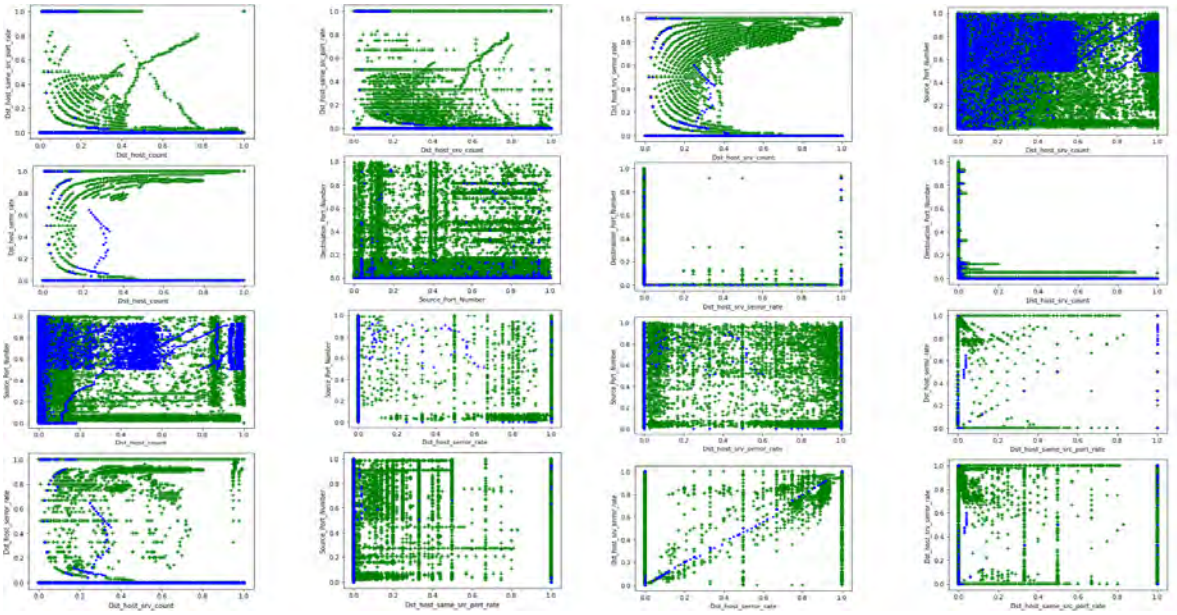


Fig. 9. Distribución de datos para algunas variables del conjunto KBD.

Dado que KBD tiene cerca de 24 variables numéricas, se realizó una comparación de la distribución de datos de variable por variable. La Fig. 8 muestra un esquema resumido de todas las comparaciones. En dicha comparación, los datos de color verde corresponden a anomalías, mientras que los de color azul se muestran sin anomalías.

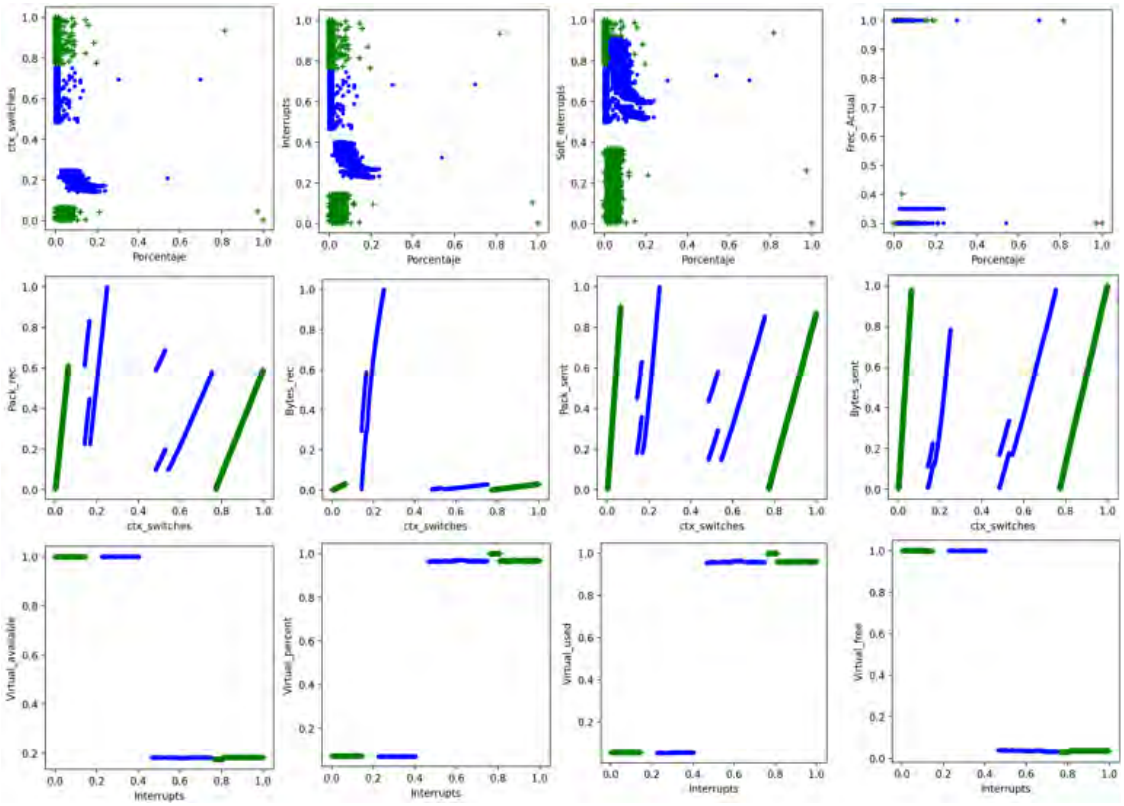


Fig. 10. Distribución de datos para algunas variables del primer conjunto de datos propio.

Lo mismo que lo anterior es mostrado para la Fig. 10, el cual corresponde a un primer conjunto de datos propio. En este conjunto de datos propio se emplearon más de 30 variables correspondientes a uso de la red zigbee, carga de CPU y RAM, a diferencia del anterior que solamente se utilizan variables en cuanto al uso de la carga de red.

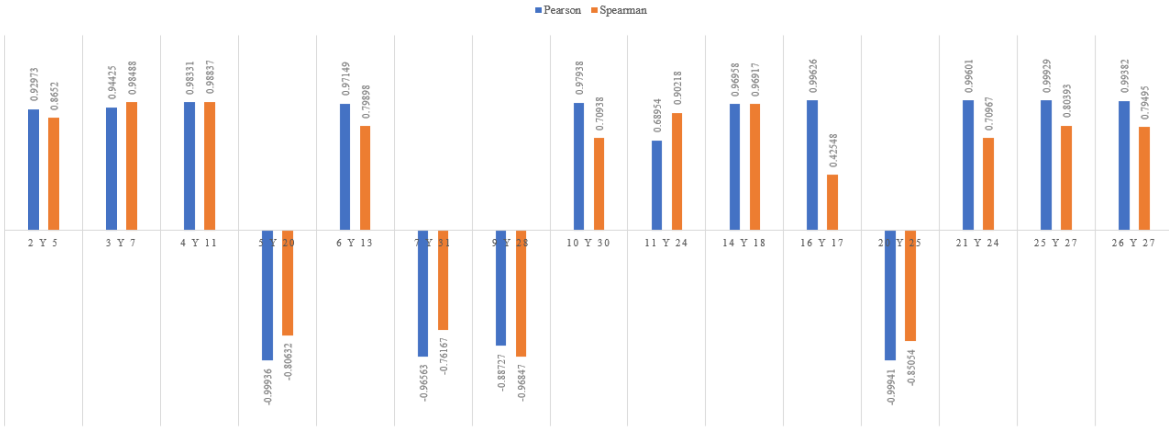


Fig. 11. Correlación de Pearson & Spearman del primer conjunto de datos propio.

Dado que una de las teorías es que el uso de CPU, RAM o red llegarían a ser dependientes unas de otras, se realizó un análisis de correlación de Pearson y Spearman. La Fig. 11 muestra que muchas variables están fuertemente relacionadas y permitiría plantear, más adelante, una nueva hipótesis.

5.1.2. Rendimiento

Las pruebas SVM con el conjunto KBD mostraban un alto rendimiento (cerca de 97% de precisión). Sin embargo, un análisis a la matriz de confusión mostró que no era capaz de clasificar los datos que no eran anómalas. La razón de la alta precisión se debía a que las clases no estaban balanceadas ni normalizadas.

```
     *****SVM Kernel Lineal C 1.00000, Gamma 0.00000:    0.9701*****
Confusion Matrix
[[84155     0]
 [ 2595     0]]
```

Fig. 12. Análisis muestra del rendimiento de SVM polinomial aplicado a KBD.

Aunque la Fig. 12 es una muestra de la matriz de confusión del kernel lineal, este resultado similar aplicaba para cada kernel.

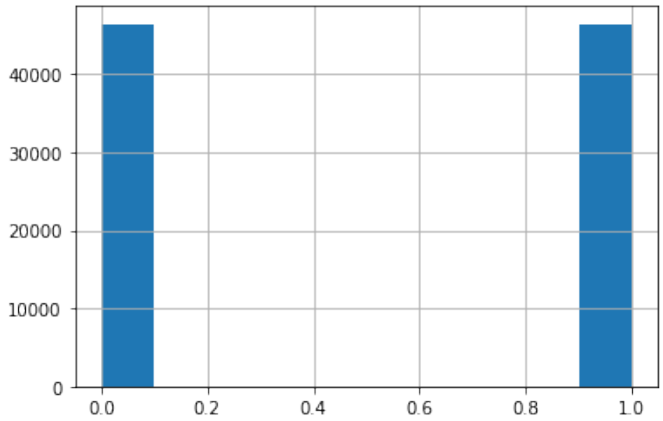


Fig. 13. Muestra de datos para KBD con clases balanceadas.

Dado que KBD se compone de 30 partes con más de 80 mil registros cada uno, se realizó un nuevo subconjunto balanceando clases. Este nuevo subconjunto contaba con más de 40 mil registros por clase (Véase Fig. 13).

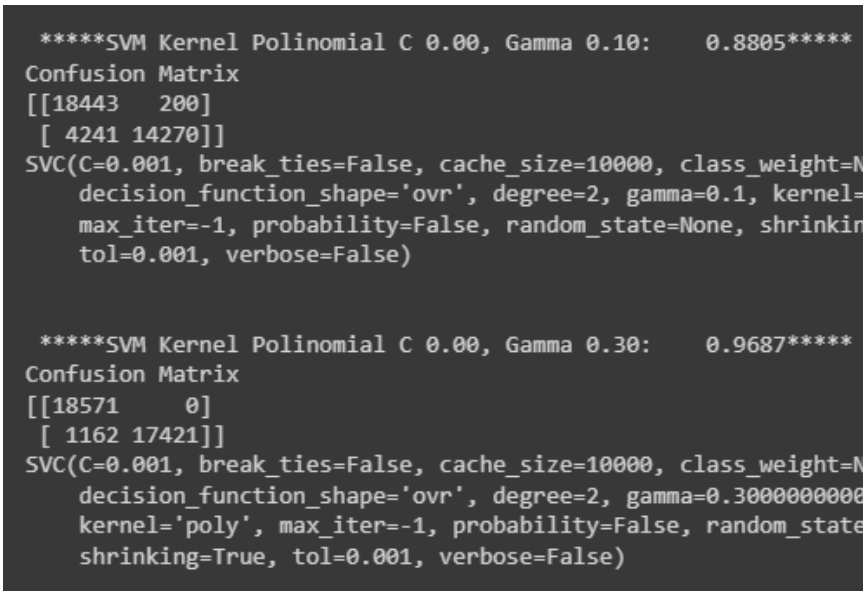


Fig. 14. Precisión y matriz de confusión de *SVM polinomial* grado 2 en KBD con clases balanceadas.

Una vez balanceados y normalizados, el rendimiento bajó desde un 88% hasta más de un 98% en los diferentes kerneles, mostrando una mejoría notoria en la matriz de confusión, tal y como se aprecia en la Fig. 14. Dicha figura es una captura del rendimiento para el Kernel polinomial, describiendo una matriz de confusión, su precisión y los hiperparámetros utilizados.

Además, en la tabla 6 se describen los mejores casos por cada kernel, así como el rendimiento obtenido.

Tabla 6. Rendimiento de SVM para el primer conjunto de datos propio.

Kernel	*C*	*gamma*	Precisión	Recall	Medida F
Linear	0.001	0.001	100	100	100
Poly 1	2.14	0.255	100	100	100
Poly 2	0.4641	0.001	100	100	100
Poly 3	1	0.004	100	100	100
Poly 4	0.021	0.001	100	100	100
Poly 5	0.633	0.354	100	100	100
Radial	2.1544	0.001	100	100	100

5.2 Etapa II.

En esta segunda etapa, las pruebas llevadas a cabo se hicieron con una reducción de variables que se explica en el capítulo 4.4.2. Con las variables reducidas, se presentan dos conjuntos de datos recolectados del mismo entorno de pruebas, pero diferentes en su constitución. El segundo contiene las mismas variables correspondientes a *cpu*, *ram* y *red*, con la diferencia

de que se agregó una columna para identificar el estado de los dispositivos (status). Además, se implementó una variante de *DoS* (*LDoS*) para obtener un conjunto de 3 clases.

De esta manera, la etapa II se divide en dos partes. La primera parte muestra, en 2 clases, la distribución de datos y el rendimiento obtenido en la clasificación de anomalías con SVM. La segunda parte muestra, en 3 clases, la distribución de datos y el rendimiento obtenido en la clasificación de estados de los dispositivos y la identificación de ataques.

5.2.1. Dos clases.

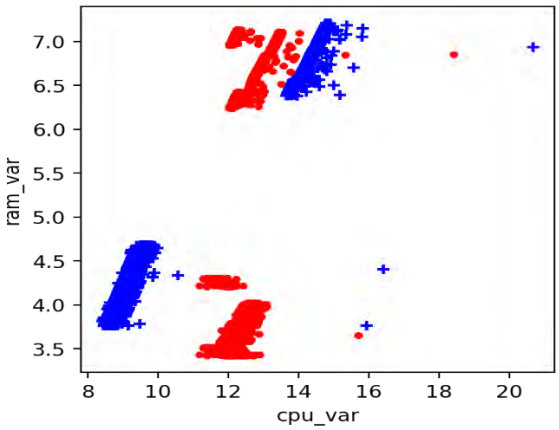


Fig. 15. Distribución de datos para el segundo conjunto de datos propio usando dos clases y dos variables.

En este apartado se muestra una distribución de datos entre las variables relacionas con *cpu* y *ram*. La idea de esto es mostrar qué tan importante son dichas variables a la hora de la identificación de intrusiones en un dispositivo que simula un vehículo no tripulado (Véase Fig. 15). En la Fig. 15, los datos en rojo representan ataques de denegación de servicio y los datos en azules de dispositivos con un comportamiento normal.

Una vez se realizaron las pruebas, siguiendo la metodología del apartado 4.4.2, se encuentra que el kernel lineal de SVM fue el menos efectivo. En su contra parte, el kernel polinolimial grado 4 fue el que mejor rendimiento presentó, con un 99.9% de precisión. La diferencia con los otros grados del mismo kernel, e incluso con el kernel radial, fueron mínimas (+/- 0.05%).

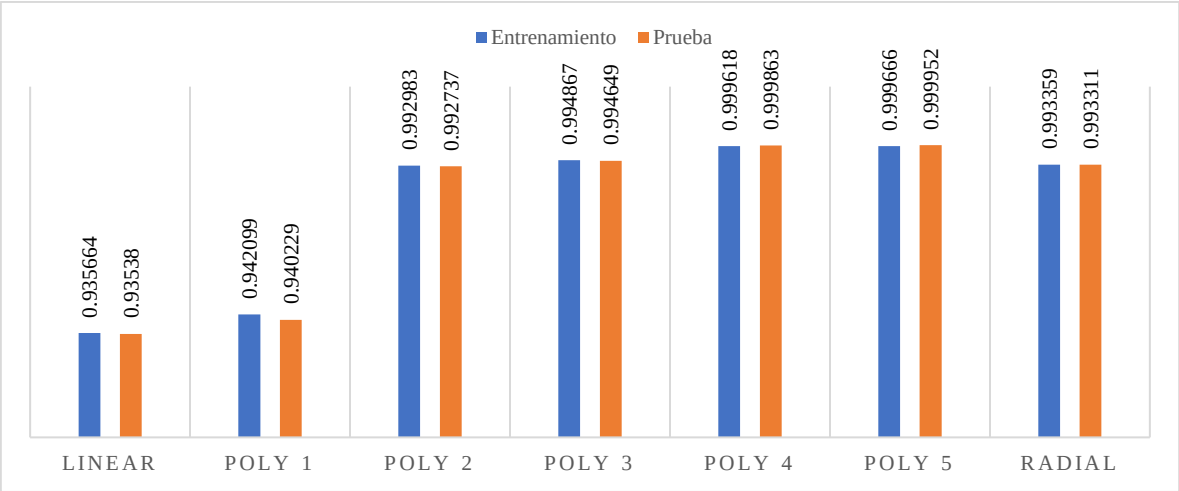


Diagrama 4. Comparativa del rendimiento SVM en el segundo conjunto de datos propio con dos clases.

El diagrama 4 muestra una comparativa entre la identificación con los datasets de entrenamiento y prueba donde el valor mínimo es 0 y el máximo posible es 1.

5.2.2. Tres clases.

En este último aparatado se realiza una reducción del conjunto a 3 variables y dos columnas de clases. El primer conjunto de pruebas con este conjunto identifica el estado de los dispositivos, teniendo como objetivo identificar si un nodo está operando de manera normal, si está siendo atacado o si está realizando un ataque. El segundo conjunto de pruebas consiste en determinar si se trata DoS, LDoS, o si en su defecto no existe anomalía alguna.

Estados

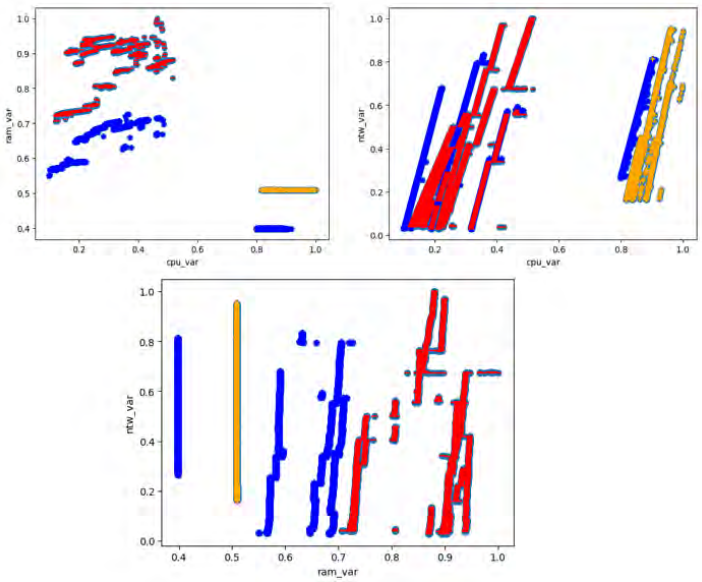


Fig. 16. Distribución de datos para el segundo conjunto de datos propios mostrando estados como clases.

La Fig. 16 muestra la distribución de datos en 3 variables en cuando a la determinación de estados de los dispositivos donde el color rojo representa a los dispositivos

atacantes, el color naranja los dispositivos atacados y los azules los comportamientos normales.

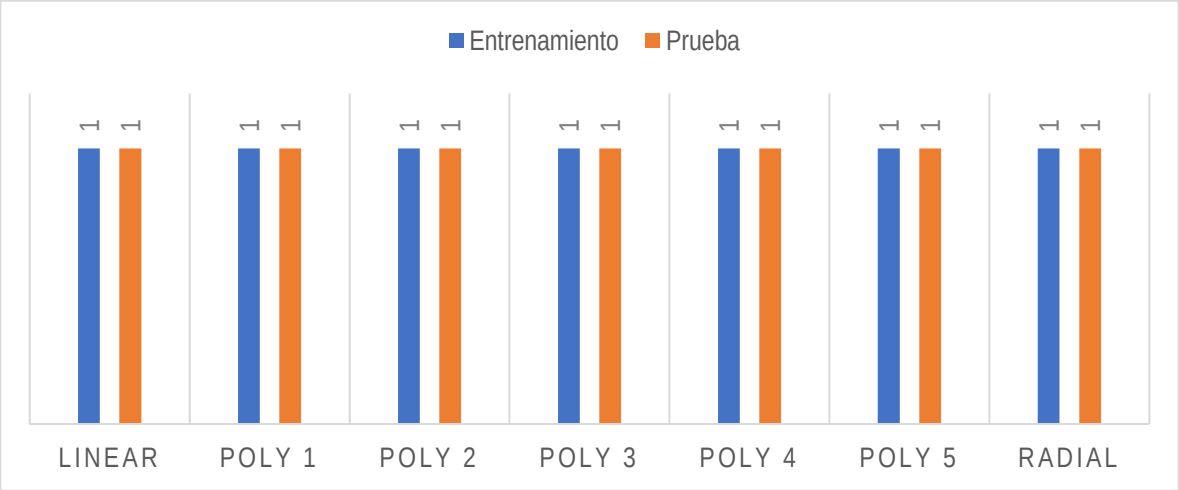


Fig. 17. Distribución de datos para el segundo conjunto de datos propio usando tres clases y tres variables.

Por su parte, la Fig. 17 muestra el rendimiento en cada uno de los *kerneles* de SVM, teniendo una identificación inequívoca de estos en cada una. Nuevamente, el gráfico es una comparativa entre conjuntos de entrenamiento y prueba donde 1 es el valor máximo posible.

Ataques

Los ataques, como se ha mencionado anteriormente, se implementaron de tipo DoS por tormenta de paquetes y una variación de tormenta de paquetes de tasa baja. El actuar de este último es enviar una ráfaga de paquetes a un nodo atacado en intervalos de segundos, y así hasta inutilizar de la red al nodo.

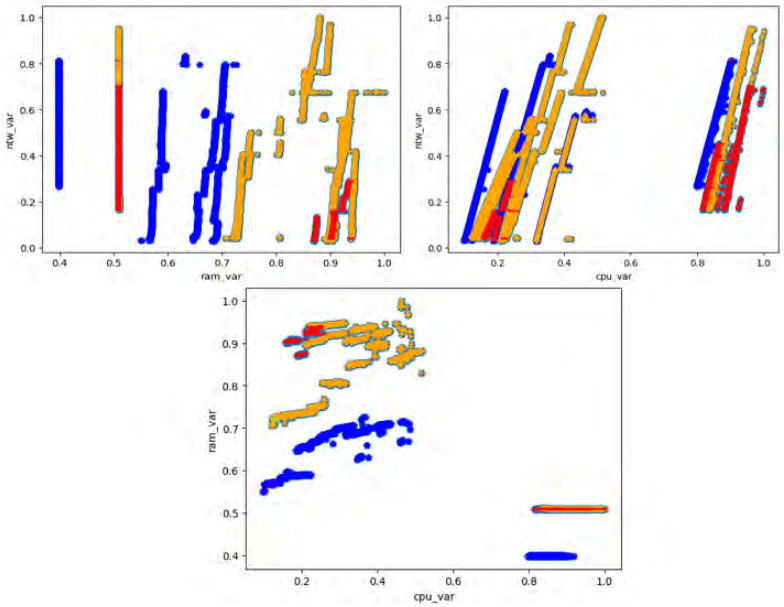


Fig. 18. Distribución de datos para el segundo conjunto de datos propios mostrando ataques como clases.

La Fig. 18 muestra una distribución de datos donde se manifiestan 2 ataques y registros normales en las 3 variables reducidas. Los datos en rojo representan los ataques DoS, los naranja los ataques LDoS y los azules dispositivos con comportamiento normal.

Tabla 7. Rendimiento en la clasificación de ataques para el segundo conjunto de datos propio.

Kernel	*Exactitud*	*Precisión*	*Especificidad*	*Sensibilidad*	*Medida F (F1 score)*	*Dispositivo*
Linear	89.70	85.56	92.39	84.04	83.72	
Poly 1	92.11	87.79	94.19	87.65	87.67	
Poly 2	99.70	99.55	99.78	99.54	99.54	
Poly 3	99.81	99.70	99.86	99.70	99.70	Raspberry pi 4
Poly 4	99.83	99.75	99.88	99.75	99.75	
Poly 5	99.79	99.68	99.84	99.68	99.68	
Radial	98.92	98.36	99.20	98.31	98.32	
Linear	100	100	100	100	100	
Poly 1	100	100	100	100	100	
Poly 2	100	100	100	100	100	
Poly 3	100	100	100	100	100	Malicioso 1
Poly 4	100	100	100	100	100	
Poly 5	100	100	100	100	100	
Radial	100	100	100	100	100	
Linear	100	100	100	100	100	
Poly 1	100	100	100	100	100	
Poly 2	100	100	100	100	100	
Poly 3	100	100	100	100	100	Malicioso 2
Poly 4	100	100	100	100	100	
Poly 5	100	100	100	100	100	
Radial	100	100	100	100	100	
Linear	100	100	100	100	100	
Poly 1	100	100	100	100	100	
Poly 2	100	100	100	100	100	
Poly 3	100	100	100	100	100	Malicioso 3
Poly 4	100	100	100	100	100	
Poly 5	100	100	100	100	100	
Radial	100	100	100	100	100	

El rendimiento mostrado en la Tabla 7 muestra que el identificador de intrusiones es completamente efectivo identificando nodos atacantes y el tipo de ataque que realizan.

Además, se ha logrado también el 99% en la identificación del ataque que se realiza a un nodo atacado.

CAPÍTULO 6. DISCUSIÓN

Este trabajo comienza con una comparación directa con KBD debido a que se ha descartado el conjunto de datos mejorado KDD. Esto fue gracias a su baja compatibilidad con el tráfico de redes IoT y, aún más en específico, con vehículos no tripulados.

KBD es un conjunto que entra en el contexto de esta aplicación y con el que este trabajo logra un rendimiento similar y hasta superior. Sin embargo, los datos recolectados se expresan en diferentes variables que precisan de servicios de red. Dichos servicios son implementados en un modelo OSI, donde el protocolo TCP/IP v4 o v6 está presente, pero este protocolo no es inherente del estándar 802.15.4 o, en su defecto, del protocolo zigbee puro. Al no ser variables propias de un entorno con zigbee puro, quedan descartadas completamente para el uso de un sistema de detección de intrusiones en un contexto de cualquiera donde se utilicen redes zigbee.

Una de las aportaciones fuertes en este trabajo consta de las nuevas variables planteadas como parte de un sistema de detección de intrusiones propio. Dicha aportación tiene que ver con el enfoque de aplicación, y que es completamente distinta del resto de trabajos publicados, donde el IDS se centra especialmente en obtener los datos directamente desde los nodos finales[7] y no del tráfico de red.

Por esta razón, la propuesta de este proyecto de tesis parte de dos bases fundamentales que no se encuentran en otros trabajos del estado del arte:

- Las variables propuestas se centran no solo en el tráfico de red, sino en el uso de memoria y el uso de procesamiento de los dispositivos.
- El sistema de detección de intrusiones se enfoca en los nodos (dispositivos finales) y no en un tráfico en general como en los trabajos estudiados.

Lo más similar a este trabajo son los sistemas de detección de intrusiones basados en agentes y son programas o ejecutables que opera directamente desde el dispositivo. Sin embargo, las limitaciones de recursos de procesamiento, memoria y red son muy remarcadas para el contexto de este trabajo (es por ello de que las funciones sean completamente operativas) y la mayoría de los contextos de aplicación IoT. Por esta razón, una de las novedades de propuesta en este trabajo reside en la utilización de un dispositivo vasto en recursos de procesamiento para la creación de n-modelos tantos n-dispositivos se contengan. Este proceso se hace 2 veces y cada modelo corresponde a la fase 2 y 3 de la arquitectura propuesta. Cada modelo SVM está construido acorde al desempeño y funcionamiento del correspondiente dispositivo final sin utilizar recursos extras en este último, solamente un programa que sirve para la entrega y recepción de los datos que alimentarán al IDS. De esta manera, se crea un sistema basado en agentes, pero que opera a través de la red,

[7] La idea surge en enfocarse directamente en cada nodo que potencialmente podría ser blanco de un ataque, asegurando directamente su ciberseguridad de manera personalizada.

convirtiéndose así en un sistema hibrido de detección de intrusiones (*HIDS* por sus siglas en inglés).

El último punto a mencionar es con respecto al método que se utiliza para hacer identificación de intrusiones y que también es menester de la aportación de esta tesis: la arquitectura. Se ha utilizado una arquitectura encontrada en Yang, Moubayed, Hamieh, & Shami (2019) y que también se ha sugerido en la mayoría de los trabajos realizados. Para este aporte, se han hecho las modificaciones necesarias para detallarlo como sigue:

Hacer un perfilamiento de cada dispositivo final. Esto implica tener datos únicos para cada dispositivo final que permitirá el entrenamiento de sus respectivos modelos.

Detección de anomalías. Antes de siquiera pensar en ataques, una columna de identificación de estados donde se establecen 3 clases: normal, nodo atacado y nodo atacante. Este concepto permitirá en un futuro enfocarse a la identificación de nodos legítimos que han sido infectados para llevar a cabo procesos maliciosos.

Detección de ataques. Al tener un modelo más que permita la identificación de ataques, se abre las puertas a un sistema de prevención de intrusiones que entre como un actor en la red y tome las medidas necesarias para salvaguardar dicho nodo de manera autónoma.

Dado que, en una red la cantidad de datos benignos son potencialmente mayores que datos malignos (momentos en los que se lleva a cabo un ataque), separar la fase 2 y 3 permite darle carga de procesamiento al HIDS únicamente cuando una anomalía se presente. Esto sucede porque la fase 3 entra en curso únicamente cuando se presenta una anomalía en la fase 2.

CAPÍTULO 7. CONCLUSIONES

A través de la revisión del estado del arte, se asevera en un 100% de los temas que los ataques de denegación de servicio y variantes son detectables analizando el tráfico en red, obteniendo hasta un 99% en la identificación y hasta un 1% en la tasa de falsos positivos, esto, empleando técnicas robustas donde se combinan técnicas de máquinas de aprendizaje, como por ejemplo, en Yang, Moubayed, Hamieh, & Shami (2019). En este documento se observa que los ataques DoS y la variante LDoS son detectables también con ayuda de variables del uso de CPU y RAM, propias de Vehiculos no tripulados. De esta manera se concluye que esta clase de ataques se manifiestan a través de estos recursos también y no solo en recursos de red como sugiere el estado del arte, haciéndolo único en su marco de aplicación y en su tipo. Los resultados con la técnica de máquinas de vector soporte en sus diferentes kerneles apoyan esta conclusión al encontrar hasta un 99% en la métrica de medida F y presentando un bajo riesgo con hasta menos del 1% en tasas de falsos positivos.

Además de esto, se logra la inferencia de dispositivos atacados y dispositivos atacantes de manera autónoma a través de SVM, demostrando una precisión del 100% para la identificación de dispositivos atacantes y hasta un 99% para la identificación de dispositivos atacados.

7.1 Trabajo futuro

Dado que se analiza un comportamiento de CPU, RAM y red, se propone implementar este modelo y arquitectura en redes convencionales, utilizando zigbee como medio de comunicación único para la monitorización de los datos.

Además, el trabajo futuro consiste en utilizar este mismo método y arquitectura para identificación de otros ataques no abordados en esta tesis.

REFERENCIAS BIBLIOGRÁFICAS

Adams, J. (01 de Junio de 2003). *Meet the Zigbee Standard*. Recuperado el Febrero de 2020, de Fierce Electronics: https://www.fierceelectronics.com/components/meet-zigbee-standard

Adams, J. (01 de Junio de 2003). *Meet the Zigbee Standard*. Recuperado el Febrero de 2020, de Fierce Electronics: https://www.fierceelectronics.com/components/meet-zigbee-standard

Ahemd, M. M., Shah, M. A., & Wahid, A. (2017). IoT security: A layered approach for attacks & defenses. *2017 International Conference on Communication Technologies (ComTech).* Rawalpindi, Pakistan: IEEE Xplore.

Ahmed, M. E., Kim, H., & Park, M. (2017). Mitigating DNS query-based DDoS attacks with machine learning on software-defined networking. *MILCOM 2017 - 2017 IEEE Military Communications Conference (MILCOM)* (págs. 2155-7586). Baltimore, MD, USA: IEEE Xplore.

Airehrour, D., Gutierrez, J., & Ray, S. K. (Mayo de 2016). Secure routing for internet of things: a survey. (ELSEVIER, Ed.) *Journal of Network and Computer Applications, 66*, 198-213. doi:https://doi.org/10.1016/j.jnca.2016.03.006

Alaba, F. A., Othman, M., Hashem, I. A., & Alotaibi, F. (15 de Junio de 2017). Internet of Things security: A survey. (ELSEVIER, Ed.) *Journal of Network and Computer Applications, 88*, 10-28. doi:https://doi.org/10.1016/j.jnca.2017.04.002

Alheeti, K. M., & McDonald-Maier, K. (2018). Intelligent intrusion detection in external communication systems for autonomous vehicles. *Systems Science & Control Engineering, 6*(1), 48-56. doi:10.1080/21642583.2018.1440260

Ali, S., Balushi, T. A., Nadir, Z., & Hussain, O. K. (2018). WSN Security Mechanisms for CPS. (C. Springer, Ed.) *Cyber Security for Cyber Physical Systems*(978-3-319-75879-4), 65 - 87. doi:https://doi.org/10.1007/978-3-319-75880-0_4

Anthi, E., Williams, L., Słowińska, M., Theodorakopoulos, G., & Burnap, P. (02 de Julio de 2019). A Supervised Intrusion Detection System for Smart Home IoT Devices. (IEE, Ed.) *IEEE Internet of Things Journal, 6*, 9042 - 9053. doi:10.1109/JIOT.2019.2926365

Atzori, L., Iera, A., & Morabito, G. (2010). The Internet of Things: A survey. *Computer Networks: The International Journal of Computer and Telecommunications Networking*, 2787-2805.

AWS. (s.f.). *¿Qué es un ataque DDoS?* Recuperado el 28 de Noviembre de 2019, de AWS Amazon: https://aws.amazon.com/es/shield/ddos-attack-protection/

Benzarti, S., Triki, B., & Korbaa, O. (2017). A Survey on Attacks in Internet of Things. En IEEE (Ed.), *2017 International Conference on Engineering & MIS (ICEMIS)*. Monastir, Tunisia. doi:10.1109/ICEMIS.2017.8273006

Birkinshaw, C., Rouka, E., & Vassilakis, V. G. (Not published). Implementing an intrusion detection and prevention system using software-defined networking: defending against port-scanning and denial-of-service attacks. *Journal of Network and Computer Applications*.

Bostani, H., & Sheikhan, M. (2017). Hybrid of anomaly-based and specification-based IDS for Internet of Things using unsupervised OPF based on MapReduce approach. *Computer Communications*, 52-71.

Brun, O., Yin, Y., & Gelenbe, E. (2018). Deep Learning with Dense Random Neural Network for Detecting Attacks against IoT-connected Home Environments. *Procedia Computer Science, 134*, 458 - 463. doi:https://doi.org/10.1016/j.procs.2018.07.183

Brunswick, U. o. (s.f.). *Canadian Institute for Cybersecurity*. Recuperado el 10 de Noviembre de 2020, de NSL-KDD dataset: https://www.unb.ca/cic/datasets/nsl.html

Chávez, G. (23 de Octubre de 2017). *Phishing cuesta a la banca mexicana 150 millones de pesos.* Recuperado el 20 de Agosto de 2019, de Expansión: https://expansion.mx/tecnologia/2017/10/23/phishing-cuesta-a-la-banca-mexicana-150-millones-de-pesos

Chen, H., Meng, C., Shan, Z., Fu, Z., & Bhargava, B. K. (08 de Marzo de 2019). A Novel Low-Rate Denial of Service Attack Detection Approach in Zigbee Wireless Sensor Network by Combining Hilbert-Huang Transformation and Trust Evaluation. (IEEE, Ed.) *IEEE Access, 7*, 32853 - 32866. doi:10.1109/ACCESS.2019.2903816

Chen, K., Zhang, S., Li, Z., Zhang, Y., Deng, Q., Ray, S., & Jin, Y. (2018). Internet-of-Things Security and Vulnerabilities: Taxonomy, Challenges, and Practice. (S. Link, Ed.) *Journal of Hardware and Systems Security, 2*, 97–110.

Chhaya, L., Sharma, P., Bhagwatikar, G., & Kumar, A. (04 de Enero de 2017). Wireless Sensor Network Based Smart Grid Communications: Cyber Attacks, Intrusion Detection System and Topology Control. (A. Vaccaro, Ed.) *Electronics, 06*. doi:https://doi.org/10.3390/electronics6010005

CISCO. (10 de Julio de 2014). *Opciones para reducir las intrusiones del falso positivo*. Recuperado el 12 de Agosto de 2020, de CISCO: https://www.cisco.com/c/es_mx/support/docs/security/firesight-management-center/117909-config-sourcefire-00.html

Cisco. (2018). *Cisco 2018 Annual Cybersecurity Report* . Cisco.

Clark, A., & Claise, B. (2011). Guidelines for Considering New Performance Metric Development. IETF.

Computer World México. (2018). *Costos ocultos de las brechas de datos aumentan los gastos para las empresas.* Recuperado el 15 de Agosto de 2019, de Noticias, Tecnología Empresarial, Seguridad: http://computerworldmexico.com.mx/costos-ocultos-de-las-brechas-de-datos-aumentan-los-gastos-para-las-empresas/

Condomines, J.-P., Zhang, R., & Larrieu, N. (2019). Network intrusion detection system for UAV ad-hoc communication: From methodology design to real test validation. *Ad Hoc Networks, 90*. doi:10.1016/j.adhoc.2018.09.004

Cusack, G., Michel, O., & Keller, E. (2018). Machine Learning-Based Detection of Ransomware Using SDN. *International Workshop on Security in Software Defined Networks & Network Function Virtualization* (págs. 1-6). Tempe, AZ, USA: ACM DIgital Library.

Deng, L., Li, D., Yao, X., Cox, D., & Wang, H. (31 de Enero de 2018). Mobile Network Intrusion detection for IoT system based on transfer learning algorithm. (S. Link, Ed.) *Cluster Computing*, 1-16. doi:https://doi.org/10.1007/s10586-018-1847-2

Deogirikar, J., & Vidhate, A. (2017). Security Attacks inIoT: A Survey. En IEEE (Ed.), *2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC).* Palladam, India. doi:10.1109/I-SMAC.2017.8058363

Dignani, J. P. (2011). *Zigbee, Análisis del protocolo.* La Plata: Universidad Nacional de La Plata .

Diro, A. A., & Chilamkurti, N. (2018). Distributed attack detection scheme using deep learning approach for Internet of Things. (ELSEVIER, Ed.) *Future Generation Computer Systems, 82*, 761 - 768. doi:https://doi.org/10.1016/j.future.2017.08.043

Elrawy, M. F., Awad, A. I., & Hamed, H. F. (04 de Diciembre de 2018). Intrusion detection systems for IoT-based smart environments: a survey. (S. Link, Ed.) *Journal of cloud computing*, 7-21. doi:https://doi.org/10.1186/s13677-018-0123-6

Fernández, R. (20 de Marzo de 2023). *Statista*. Obtenido de El Internet de las cosas (IoT) - Datos estadísticos: https://es.statista.com/temas/6976/el-internet-de-las-cosas-iot/#topicOverview

Frenzel, L. (22 de Marzo de 2013). *What's The Difference Between IEEE 802.15.4 And Zigbee Wireless?* Obtenido de ElectronicDesign: https://www.electronicdesign.com/unused/article/21796046/whats-the-difference-between-ieee-802154-and-zigbee-wireless

Frustaci, M., Pace, P., Aloi, G., & Fortino, G. (2018). Evaluating Critical Security Issues of the IoT World: Present and Future Challenges. *IEEE Internet of Things Journal*, 2483 - 2495.

Fuller, M., Jenkins, M., & Tjølsen, K. (2017). Security Analysis of the August Smart Lock. (M. I. Technology, Ed.) Obtenido de https://courses.csail.mit.edu/6.857/2017/project/3.pdf

Future, M. R. (Mayo de 2020). *Zigbee Market Research Report - Forecast to 2023*. Obtenido de Market Research Future: https://www.marketresearchfuture.com/reports/zigbee-market-2617

Gajewski, M., Batalla, J. M., Mastorakis, G., & Mavromoustakis, C. X. (2019). A distributed IDS architecture model for Smart Home systems. (S. Link, Ed.) *Cluster Computing, 22,* 1739–1749. doi:https://doi.org/10.1007/s10586-017-1105-z

Gendreau, A. A., & Moorman, M. (2016). Survey of Intrusion Detection Systems towards an End to End Secure Internet of Things. *2016 IEEE 4th International Conference on Future Internet of Things and Cloud (FiCloud).* Vienna, Austria: IEEE Xplore.

Go socket. (s.f.). *EL ROBO DE IDENTIDAD COSTÓ 118 MILLONES DE PESOS.* Recuperado el 28 de Agosto de 2019, de http://iofacturo.mx/economia/el-robo-de-identidad-costo-118-millones-de-pesos

Go Socket The Companies Network. (2015). *EL ROBO DE IDENTIDAD COSTÓ 118 MILLONES DE PESOS*. Obtenido de https://iofacturo.mx/economia/el-robo-de-identidad-costo-118-millones-de-pesos

Goyal, K. K., Garg, A., Rastogi, A., & Singhal, S. (2018). A Literature Survey on Internet of Things (IoT). *Int. J. Advanced Networking and Applications, 09*(06), 3663-3668. Obtenido de http://oaji.net/articles/2017/2698-1528118826.pdf

HaddadPajouh, H., Parizi, R., Dehghantanha, A., Aledhari, M., & Karimipour, H. (2019). A Survey on Internet of Things Security: Requirements, Challenges, and Solutions. (Elsevier, Ed.) *Internet of Things*(100129). doi:https://doi.org/10.1016/j.iot.2019.100129

Hasrouny, H., Samhat, A. E., Bassil, C., & Laouiti, A. (2017). VANet security challenges and solutions: A survey. *Vehicular Communications, 7*, 7-20. doi:http://dx.doi.org/10.1016/j.vehcom.2017.01.002

Hewlett-Packard-Enterprise. (s.f.). *¿Qué es el Internet de las cosas (IoT)?* Recuperado el 07 de Agosto de 2023, de ¿Qué es el Internet de las cosas (IoT)?: https://www.hpe.com/mx/es/what-is/internet-of-things-iot.html

Ingeniería, R. A. (s.f.). *Real Academia de Ingeniería*. Recuperado el 17 de Abril de 2020, de Real Academia de Ingeniería: http://diccionario.raing.es/es/lema/calidad-de-servicio-0

Jain, R. (02 de 04 de 2019). *circuitdigest.com*. Obtenido de How to Interface XBee Module with Raspberry Pi: https://circuitdigest.com/microcontroller-projects/raspberry-pi-xbee-module-interfacing

Jan, S. U., Ahmed, S., Shakhov, V., & Koo, I. (2019). Toward a Lightweight Intrusion Detection System for the Internet of Things. *IEEE Access (Volume: 7)*, 42450 - 42471.

Jiang, F., Fu, Y., Gupta, B. B., Lou, F., Rho, S., Meng, F., & Tian, Z. (2018). Deep Learning based Multi-channel intelligent attack detection for Data Security. *IEEE Transactions on Sustainable Computing*, 1-1.

Jokar, P., & Leung, V. C. (2018). Intrusion Detection and Prevention for Zigbee-Based Home Area Networks in Smart Grids. *IEEE Transactions on Smart Grid*, 1800 - 1811.

Justin, V., Marathe, N., & Dongre, N. (2017). Hybrid IDS using SVM classifier for detecting DoS attack in MANET application. *2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*. Palladam, India.

Kaspersky. (s.f.). *¿Qué es la Internet de las cosas? Definición y explicación*. Recuperado el 05 de Agosto de 2023, de https://latam.kaspersky.com/resource-center/definitions/what-is-iot

Khatoun, R., & Zeadally, S. (13 de Marzo de 2017). Cybersecurity and Privacy Solutions in Smart Cities. (IEEE, Ed.) *IEEE Communications Magazine , 55*, 51-59. doi:10.1109/MCOM.2017.1600297CM

Kim, H. K. (s.f.). *CAR-HACKING DATASET*. Recuperado el 02 de Enero de 2021, de Car-Hacking Dataset for the intrusion detection: https://ocslab.hksecurity.net/Datasets/CAN-intrusion-dataset

Kolias, C., Kambourakis, G., Stavrou, A., & Voas, J. (2017). DDoS in the IoT: Mirai and Other Botnets. (IEEE, Ed.) *Computer, 50*(0018-9162), 80 - 84. doi:10.1109/MC.2017.201

Lopez-Martin, M., Carro, B., Sanchez-Esguevillas, A., & Lloret, J. (2017). Conditional Variational Autoencoder for Prediction and Feature Recovery Applied to Intrusion Detection in IoT. *Sensors*.

Lu, Y., & Xu, L. D. (2019). Internet of Things (IoT) Cybersecurity Research: A Review of Current Research Topics. (IEEE, Ed.) *IEEE Internet of Things Journal, 6*(18653874), 2103 - 2115. doi:10.1109/JIOT.2018.2869847

Mao, B., Tang, F., Fadlullah, Z. M., Kato, N., Akashi, O., Inoue, T., & Mizutani, K. (2018). A Novel Non-Supervised Deep-Learning-Based Network Traffic Control Method for Software Defined Wireless Networks. *IEEE Wireless Communications*, 74-81.

Microsoft. (14 de Noviembre de 2019). *Solución de problemas del firewall de aplicaciones web (WAF) de Azure Application Gateway*. Recuperado el 02 de Agosto de 2020, de Azure: https://docs.microsoft.com/es-es/azure/web-application-firewall/ag/web-application-firewall-troubleshoot

Mustapha, H., & Alghamdi, A. M. (2018). DDoS attacks on the internet of things and their prevention methods. *ICFNDS '18 Proceedings of the 2nd International Conference on Future Networks and Distributed Systems.* Amman, Jordan: ACM Digital Library.

Nawir, M., Amir, A., Yaakob, N., & Lynn, O. B. (2016). Internet of Things (IoT): Taxonomy of security attacks. *2016 3rd International Conference on Electronic Design (ICED).* Phuket, Thailand: IEEE Xplore.

Onofre, J. S. (05 de Mayo de 2017). *México entrará al Top 5 en AL del Internet de las cosas en 2020*. Recuperado el 15 de Agosto de 2019, de www.eleconomista.com.mx: https://www.eleconomista.com.mx/tecnologia/Mexico-entrara-al-Top-5-en-AL-del-Internet-de-las-cosas-en-2020-20170505-0019.html

Pecorella, T., Pierucci, L., & Nizzi, F. (2018). "Network Sentiment" Framework to Improve Security and Privacy for Smart Home. *Future Internet*.

Pereira, D. S., Morais, M. R., Nascimento, L. B., Alsina, P. J., Santos, V. G., Fernandes, D. H., & Silva, M. R. (23 de Marzo de 2020). zigbee Protocol-Based Communication Network for Multi-Unmanned Aerial Vehicle Networks. (IEEE, Ed.) *IEEE Access*, 57762 - 57771. doi:10.1109/ACCESS.2020.2982402

Porto, J. P., & Gardey, A. (2008). *Concepto de seguridad*. Obtenido de definicion.de: https://definicion.de/seguridad/

Procopiou, A., Komninos, N., & Douligeris, C. (3 de Febrero de 2019). ForChaos: Real time application DDoS detection using Forecasting and Chaos Theory in Smart Home IoT Network. (Hindawi, Ed.) *Wireless Communications and Mobile Computing, 2019*. doi:https://doi.org/10.1155/2019/8469410

Rani, C., Modares, H., Sriram, R., Mikulski, D., & Lewis, F. L. (2015). Security of unmanned aerial vehicle systems against cyber-physical attacks. (JDMS, Ed.) *Journal of Defense Modeling and Simulation: Applications, Methodology, Technology*, 1-12. doi:10.1177/1548512915617252

Raza, S., Wallgren, L., & Voigt, T. (Noviembre de 2013). SVELTE: Real-time intrusion detection in the Internet of Things. *Ad Hoc Networks*, 2661-2674. doi:https://doi.org/10.1016/j.adhoc.2013.04.014

RedHat. (20 de Enero de 2023). *¿Qué es el Internet de las cosas (IoT)?* Recuperado el 07 de Julio de 2023, de ¿Qué es el Internet de las cosas (IoT)?: https://www.redhat.com/es/topics/internet-of-things/what-is-iot

RedHat. (20 de Enero de 2023). *¿Qué es el Internet de las cosas (IoT)?* Recuperado el 27 de Marzo de 2023, de ¿Qué es el Internet de las cosas (IoT)?: https://www.redhat.com/es/topics/internet-of-things/what-is-iot

Remírez, D. (06 de Junio de 2018). *Ataques de malware pueden costarle hasta 2.5 mdd a las empresas.* Recuperado el 21 de Agosto de 2019, de Forbes:

https://www.forbes.com.mx/ataques-de-malware-pueden-costarle-hasta-2-5-mdd-a-las-empresas/

Riquelme, R. (10 de Abril de 2019). *Pérdida de datos le cuesta a una empresa en México más de 1 millón de dólares: DELL EMC.* Obtenido de El Economista: https://www.eleconomista.com.mx/tecnologia/Perdida-de-datos-le-cuesta-a-una-empresa-en-Mexico-mas-de-1-millon-de-dolares-DELL-EMC-20190410-0079.html

Shone, N., Ngoc, T. N., Phai, V. D., & Shi, Q. (23 de Enero de 2018). A Deep Learning Approach to Network Intrusion Detection. (IEEE, Ed.) *IEEE Transactions on Emerging Topics in Computational Intelligence, 02*(1), 41 - 50. doi:10.1109/TETCI.2017.2772792

Siboni, S., Shabtai, A., & Elovici, Y. (2018). Leaking data from enterprise networks using a compromised smartwatch device. *SAC '18: Proceedings of the 33rd Annual ACM Symposium on Applied Computing* (págs. 741 - 750). ACM Digital. doi:https://doi.org/10.1145/3167132.3167214

Sicari, S., Rizzardi, A., Miorandi, D., & Coen-Porisini, A. (04 de Junio de 2018). REATO: REActing TO Denial of Service attacks in the Internet of Things. (ELSEVIER, Ed.) *Computer Networks, 137*, 17 - 48. doi:https://doi.org/10.1016/j.comnet.2018.03.020

Sonar, K., & Upadhyay, H. (2014). A Survey: DDOS Attack on Internet of Things. *International Journal of Engineering Research and Development*, 58-63.

Song, H. M., Woo, J., & Kim, H. K. (2020). In-vehicle network intrusion detection using deep convolutional neural network. (ELSEVIER, Ed.) *Vehicular Communications, 21*. doi:10.1016/j.vehcom.2019.100198

Song, J., Takakura, H., & Okabe, Y. (s.f.). *Description of Kyoto University Benchmark Data.* Kyoto: Kyoto University.

Statista. (2023). *Dispositivos conectados (Internet de las cosas) a nivel mundial de 2015 a 2027(en miles de millones de unidades)*. Obtenido de Statista: https://es.statista.com/estadisticas/517654/prevision-de-la-evolucion-de-los-dispositivos-conectados-para-el-internet-de-las-cosas-en-el-mundo/

Statista. (Enero de 2023). *Size of the global autonomous vehicle market in 2021 and 2022, with a forecast through 2030(in million U.S. dollars)*. Recuperado el 07 de Marzo de 2023, de Size of the global autonomous vehicle market in 2021 and 2022, with a forecast through 2030(in million U.S. dollars): https://www.statista.com/statistics/1224515/av-market-size-worldwide-forecast/

Statista Research Department. (27 de Noviembre de 2016). *www.statista.com*. Recuperado el 15 de Agosto de 2019, de www.statista.com: https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/

Statista Research Department. (2020). *Internet of Things (IoT) connected devices installed base worldwide from 2015 to 2025(in billions)*. Recuperado el 2020, de Internet of Things (IoT) connected devices installed base worldwide from 2015 to 2025(in billions): https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/

Takase, H., Kobayashi, R., Kato, M., & Ohmura, R. (2019). A prototype implementation and evaluation of the malware detection mechanism for IoT devices using the processor information. *International Journal of Information Security*, 1 - 11.

Tweneboah-Koduah, S., Skouby, K. E., & Tadayoni, R. (2017). Cyber security threats to IoT applications and service domains. *Wireless Personal Communications*, 169–185.

Wikipedia. (16 de Junio de 2019). *Falso positivo (informática)*. Recuperado el 20 de Agosto de 2020, de Wikipedia la enciclopedia libre: https://es.wikipedia.org/wiki/Falso_positivo_(inform%C3%A1tica)#:~:text=Un%20falso%20positivo%20en%20inform%C3%A1tica,es%20ning%C3%BAn%20virus%20o%20malware.

Williams, P., Rojas, P., & Bayoumi, M. (2019). Security Taxonomy in IoT – A Survey. En IEEE (Ed.), *2019 IEEE 62nd International Midwest Symposium on Circuits and Systems (MWSCAS)*. Dallas, Tx, USA. doi:10.1109/MWSCAS.2019.8884913

Wong, J. C., & Solon, O. (12 de Mayo de 2017). *Massive ransomware cyber-attack hits nearly 100 countries around the world.* Recuperado el 15 de Agosto de 2019, de The Guardian: https://www.theguardian.com/technology/2017/may/12/global-cyber-attack-ransomware-nsa-uk-nhs

Wood, D., Apthorpe, N., & Feamster, N. (27 de Marzo de 2018). *Cleartext Data Transmissions in Consumer IoT Medical Devices.* Obtenido de https://arxiv.org/pdf/1803.10147.pdf

Yaacoub, J.-P., Noura, H., Salman, O., & Chehab, A. (Septiembre de 2020). Security analysis of drones systems: Attacks, limitations, and recommendations. (ELSEVIER, Ed.) *Internet of Things, 11*. doi:10.1016/j.iot.2020.100218

Yang, L., Moubayed, A., Hamieh, I., & Shami, A. (2019). Tree-based Intelligent Intrusion Detection System in Internet of Vehicles. En IEEE (Ed.), *2019 IEEE Global Communications Conference (GLOBECOM)*. Waikoloa, HI, USA, USA. doi:10.1109/GLOBECOM38437.2019.9013892

Yang, Q., & Huang, L. (2018). Zigbee Technology. En Q. Yang, L. Huang, & S. Springer (Ed.), *Inside Radio: An Attack and Defense Guide* (págs. 227-265). Singapore: Springer. doi:10.1007/978-981-10-8447-8_7

Ye, M., Jiang, N., Yang, H., & Yan, Q. (2017). ecurity analysis of Internet-of-Things: A case study of august smart lock. *2017 IEEE Conference on Computer Communications*

Workshops (INFOCOM WKSHPS). Atlanta, GA, USA: IEEE. doi:10.1109/INFCOMW.2017.8116427

Zaidan, A. A., Zaidan, B. B., Qahtan, M. Y., Albahri, O. S., Albahri, A. S., Alaa, M., . . . Lim, C. K. (07 de Marzo de 2018). A survey on communication components for IoT-based technologies in smart homes. (S. Link, Ed.) *Telecommunication Systems, 69*, 1-25. doi:https://doi.org/10.1007/s11235-018-0430-8

Zardari, Z. A., He, J., Zhu, N., Mohammadani, K. H., Pathan, M. S., Hussain, M. I., & Memon, M. Q. (05 de Marzo de 2019). A Dual Attack Detection Technique to Identify Black and Gray Hole Attacks Using an Intrusion Detection System and a Connected Dominating Set in MANETs. (MDPI, Ed.) *Future Internet*. doi:doi:10.3390/fi11030061

Zheng, L., Yuan, H., Peng, X., Zhu, G., Guo, Y., Xu, H., & Deng, G. (2019). Research on Distributed High Speed Network Intrusion Prevention System. *CSIA 2019: Cyber Security Intelligence and Analytics* (págs. 1118-1126). Switzerland: Springer, Cham. doi:https://doi.org/10.1007/978-3-030-15235-2_148