UNIVERSIDAD SANTO TOMÁS

DESARROLLO DE TÉCNICA DE CIBERSEGURIDAD PARA LA ENCRIPTACIÓN DE DATOS Y DETECCIÓN DE ANOMALÍAS EN LA COMUNICACIÓN DE UN PLC CON LA NUBE EN LA INDUSTRIA 4.0

Realizado por

María Alejandra Barrios Villalobos

Miguel Alberto Esteban López

Un proyecto enviado en cumplimiento del requisito parcial
para optar por el grado de Ingeniero Electrónico

Grupo de Investigación GED (Grupo de Estudio y Desarrollo en Robótica)
Facultad de Ingeniería Electrónica
División de Ingenierías
Bogotá

Septiembre de 2021

DESARROLLO DE TÉCNICA DE CIBERSEGURIDAD PARA LA ENCRIPTACIÓN DE DATOS Y DETECCIÓN DE ANOMALÍAS EN LA COMUNICACIÓN DE UN PLC CON LA NUBE EN LA INDUSTRIA 4.0

Realizado por

María Alejandra Barrios Villalobos

Miguel Alberto Esteban López

Un proyecto enviado en cumplimiento del requisito parcial
para optar por el grado de Ingeniero Electrónico

Dirigido por

Ing. Armando Mateus Rojas, MSc

Co-Dirigido por

Ing. Edgar Camilo Camacho Poveda, MSc

Jurados

Ing. José Luis Paternina Durán, MEd

Ing. Maribel Anaya Veja, PhD

Grupo de Investigación GED (Grupo de Estudio y Desarrollo en Robótica)
Facultad de Ingeniería Electrónica
División de Ingenierías
Bogotá

Septiembre de 2021

DECLARACIÓN DE AUTORÍA

Nosotros, **MARÍA ALEJANDRA BARRIOS VILLALOBOS** con **C.C. Nº 1.018.505.557** y **MIGUEL ALBERTO ESTEBAN LOPEZ** con **C.C 1.032.504.504,** declaramos que el proyecto de grado denominado **"DESARROLLO DE TÉCNICA DE CIBERSEGURIDAD PARA LA ENCRIPTACIÓN DE DATOS Y DETECCIÓN DE ANOMALÍAS EN LA COMUNICACIÓN DE UN PLC CON LA NUBE EN LA INDUSTRIA 4.0"**, se ha desarrollado de manera íntegra, respetando derechos intelectuales de las personas que han desarrollado conceptos mediante las citas en las cuales indican la autoría, y cuyos datos se detallan de manera más completa en la bibliografía. En virtud de esta declaración, nos responsabilizamos del contenido, autenticidad y alcance del presente proyecto.

Bogotá, Julio 16 de 2021

MARÍA ALEJANDRA BARRIOS
VILLALOBOS
C.C. 1.018.505.557

MIGUEL ALBERTO ESTEBAN LOPEZ
C.C. 1.032.504.504

Abstract

This project presents a contribution to the cybersecurity of companies that are migrating their services to the Industry 4.0 in Colombia. The presented technique is intended to give greater protection to the data that passes through the communication between a PLC and the cloud by making use of blackchain-based encryption and artificial intelligence. The results show that this technique is functional and that it provides greater security fot the company's information.

Resumen

El presente trabajo de grado presenta un aporte a la ciberseguridad de las empresas que están migrando sus servicios hacia la Industria 4.0 en Colombia. Con la técnica presentada se pretende dar mayor protección a los datos que pasan por la comunicación entre un *PLC* y la nube haciendo uso de la encriptación basada en blockchain y una inteligencia artificial. Los resultados muestran que esta técnica es funcional y que proporciona mayor seguridad para la información de las empresas.

Agradecimientos

Agradezco a Dios, mis padres Sandra Villalobos y Luis Barrios, y familiares por apoyarme en todo momento. A Juan Pablo Romero por ser un pilar importante en mi vida y para el desarrollo exitoso de este proyecto. También a mis mejores amigos por siempre formar un gran equipo y estar allí cuando más nos necesitamos. A los directores de este proyecto de grado por guiarnos en su desarrollo y su disposición en todo momento. Finalmente, a mi compañero de proyecto de grado por su trabajo para lograr culminar este proyecto y ser un buen amigo.

María Alejandra Barrios Villalobos

Agradezco a Dios y mis papás Edgar Esteban Alonso y Marcela López Landinez por darme la vida y todo lo que tengo, a mis hermanos Andrés Felipe y Leonardo, a mi cuñada Katherine Valderrama que han sido un pilar fundamental en mi vida personal y profesional, a mi sobrino Alejandro por hacerme pensar en el mundo del mañana, y cómo olvidar a Pedro Pablo Salamanca Moncada, Jorge Rangel Pinzón y Rafael Tovar Camacho, que de manera indirecta han inspirado este trabajo por medio de su apoyo y consejos. Gracias Armando Mateus y Edgar Camilo Camacho por el trabajo, tiempo y consejos que permitieron llevar a cabo este trabajo de grado. Y por ultimo a María Alejandra Barrios muchísimas gracias por acompañarme estos años, por ser una excelente amiga y compañera de trabajo.
Dios los bendiga.

Miguel Alberto Esteban López

Glosario

- **CIM:** Manufactura integrada con computador.
- **CSV:** Valores separados por coma.
- **DFEL:** Aprendizaje integrado de funciones profundas.
- **DGA:** Algoritmo para la generación de dominios.
- **GRBM:** Máquina de Boltzman gausiana binaria restringida.
- **HMM:** Modelos ocultos de Markov.
- **IoT:** Internet de las cosas.
- **IP:** Protocolo de internet.
- **LSTM:** Memoria de largo a corto plazo (*Long Short-Term Memory*).
- **Malware:** Software malicioso.
- **MinTIC:** Ministerio de las tecnologías de la información y comunicaciones de la República de Colombia.
- **NLP:** Procesamiento de lenguaje natural.
- **PCA:** Análisis de componentes principales.
- **PLC:** Controlador lógico programable.
- **POW:** Prueba de trabajo.
- **RBM:** Máquina de Boltzman binaria restringida.
- **Pymes:** Pequeñas y medianas empresas.
- **RNN:** Red neuronal recurrente.
- **SCADA:** Supervisión, control y adquisición de datos.
- **SDIIoT:** Redes definidas por software para internet de las cosas industrial.
- **SDN:** Redes definidas por software.
- **SIEM:** Gestión de información y eventos de seguridad.
- **SPF:** Punto único de fallo.

- **TCP:** Protocolo de control de transferencia.
- **TI:** Tecnologías de la información.
- **USB:** Bus universal en serie.

Índice general

Índice de cuadros

Índice de figuras

Capítulo 1

Preámbulo

1.1. Introducción

Desde sus inicios, la humanidad se ha visto en la necesidad de crear productos para suplir sus necesidades. Desde el siglo XIX, la creación de estos productos se empezó a realizar a través de la industria, la cual ha crecido hasta tal punto de convertirse en una de las bases de la economía a nivel mundial evolucionando hacia grandes procesos de industrialización con alcance global y distintos sectores productivos [1]. Para esto, la industria ha pasado por varias revoluciones tecnológicas a lo largo de la historia, las cuales se han generado como consecuencia de la implementación de diferentes tecnologías innovadoras, desde la máquina a vapor hasta el Internet de las cosas (IoT por sus siglas en inglés) [2].

Actualmente, la industria se encuentra en su cuarta revolución, la cual consiste en integrar la automatización e intercambio de datos con IoT, los sistemas ciberfísicos y la computación en la nube. Esta unión trae ventajas para las diferentes empresas de la industria, destacando la agilización de los procesos de producción, flexibilidad y alta disponibilidad de los servicios que se consumen en la nube a partir de las necesidades y capacidades de la organización, monitoreo y control remoto de una planta industrial en tiempo real, entre otros [3].

Así mismo, esta revolución también puede traer desventajas para la industria, como, vulnerabilidad generada por la conectividad de algunos equipos industriales que anteriormente operaban de forma aislada (*Stand Alone*) y que con esta revolución van a contar con conexión a Internet, lo cual puede conllevar a la corrupción y robo de información. Es por esto que surge la necesidad de crear e implementar técnicas para la protección de la información generada en los procesos industriales [4].

1.2. Planteamiento del problema

Con la adopción de nuevas tecnologías surgen riesgos de que estas sean utilizadas de forma negativa por individuos para obtener beneficios o afectar a una o varias personas. Vandalismo informático, robo, cibercrimen o negocios oscuros, han sido el origen de muchos *malwares* (programa malicioso, diseñado para causar daños a un equipo o dispositivo electrónico) [5]. Diversas empresas en Colombia han experimentado las consecuencias de los ciberataques que han perjudicado sus operaciones ocasionando pérdidas desde un millón de pesos hasta los 4.000 millones de pesos [6]. La razón para ser objetivos de ataques cibernéticos, se debe al uso de dispositivos electrónicos en sus operaciones acompañado de la falsa idea de seguridad por parte de las empresas, ocasionando una falta de interés en invertir en los procesos y equipos necesarios para proteger la integridad de los sistemas de información [7]. Este problema no solo debe ser afrontado por grandes empresas, ya que en un estudio de los Laboratorios Kaspersky, 43 % de los ciberataques son dirigidos a las pymes, de las cuales el 60 % desaparecen después de haber sufrido un ciberataque[8].

A nivel industrial sucede algo similar, pues en la actualidad dicho campo se encuentra en su cuarta revolución, conocida como industria 4.0, la cual consiste en conectar los dispositivos de nivel SCADA (control de procesos) a la nube, mejorando de esta forma la comunicación entre los dispositivos que conforman sistemas automatizados, para el aprovechamiento de los beneficios que ha traído Internet a los procesos industriales y a su vez permitir realizar cambios en estos de forma rápida [3]. El desarrollo de proyectos y sistemas de automatización se lleva a cabo con base en el modelo de la pirámide de automatización CIM (*Computer Integrated Manufactured*, Manufactura Integrada por Computador), en la que el PLC (*Programmable Logic Controller*, o Controlador Lógico Programable en español) constituye el componente principal de la capa de control para dispositivos de campo, consolidando la operación de sensores y actuadores. En la industria 4.0 el PLC no se encuentra físicamente conectado con los dispositivos de campo sino que lo está de forma lógica; esto requiere que tanto el PLC como los dispositivos de campo y los sistemas SCADA (que se encuentran en un nivel superior al PLC) tengan que comunicarse a través de servicios en la nube. Este método de comunicación posibilita el riesgo de sufrir un ciberataque, por lo que se desea saber cómo se protegen las empresas de actividad industrial de estos ataques a través de la identificación de un conjunto de amenazas de ciberataque a las que se enfrenta un PLC en la Industria 4.0 haciendo uso de tecnologias basadas en *Blockchain* e inteligencia artificial, para así lograr responder a la pregunta de investigación ¿Cómo desarrollar una técnica para la encriptación y detección de anomalías en la comunicación entre un PLC y la nube para identificar y prevenir ciberataques que afecten la actividad

productiva de un sistema de automatización en la industria 4.0?

1.3. Justificación

El surgimiento de la industria 4.0 trae para Colombia un desafío de seguridad tecnológica asociado a la migración de sistemas y procesos aislados de automatización (tradicional) hacia la automatización basada en la nube. La industria nacional ha optado tradicionalmente por desarrollar sus procesos de automatización con sistemas aislados basados en PLC, esto debido a factores tales como confiabilidad y relación costo beneficio. Sin embargo, las nuevas tendencias implican un desafío relacionado con la actualización hacia la industria 4.0, en las que se exige la implementación no solo de la infraestructura de la conectividad de la planta, sino con los mecanismos de seguridad correspondientes a la protección e integridad de los datos, así como el MinTIC (ministerio de las tecnologías de la información y comunicaciones de Colombia) da recomendaciones de buenas practicas para la privacidad y protección de datos en servicios en Internet [9], es necesario contemplar mecanismos adaptados al entorno de industria 4.0. Estos mecanismos de seguridad son aquellos que permiten que la red de una fábrica no sea afectada por un ataque cibernético, como pueden ser *Stuxnet* o *Flame*, entre otros.

La industria 4.0 fue definida en el año 2011 por el gobierno alemán como "un impulso para para cambiar el sector de fabricación en la automatización tecnológica"[3], es decir, que la cuarta revolución industrial es la transición de los procesos actuales hacia tecnologías digitales. Esto significa que el escenario de seguridad ha cambiado, volviéndose más comunes los ciberataques, siendo más sofisticados y un punto débil en la seguridad nacional de los países [4].

Entre las técnicas y conceptos tradicionales para cuidar los datos y la integridad de los sistemas que se venían implementando en plantas con sistemas automatizados se encuentra el mantener aislados los equipos de Internet, hacer uso de dispositivos de almacenamiento nuevos para el intercambio de información entre los sistemas de la planta y la creencia de que los detalles técnicos de equipos industriales son conocidos por los expertos y operarios del sistema; estas técnicas probaron ser funcionales durante los años en que la operación aislada fue el paradigma predominante del sector. Sin embargo, al no contemplar que los expertos en ciberseguridad de las empresas prestaran su atención en los sistemas automatizados, dichas técnicas resultaron poco confiables para los nuevos retos, como lo demostraron *Stuxnet* y *Flame* [10].

Stuxnet es un gusano que se infiltró en la red de la planta nuclear iraní de Natanz, siendo así que saboteó la programación de los PLC de las centrifugadoras de la planta, ocasionando la

destrucción de mil (1.000) máquinas centrifugadoras luego de alterar sus velocidades de giro [11]. *Stuxnet* no es un *malware* cualquiera, es considerado una ciberarma (pieza de software o dispositivo, que se utiliza para realizar un ataque a un sistema [12]) contra sistemas industriales que muy probablemente fue financiado por algún Estado [13].

El caso de *Stuxnet* es muy importante, porque las plantas iraníes a pesar de que manejaban los sistemas de manera aislada, no pudieron evitar ser blanco de un ciberataque que afectó directamente a los sistemas de control. Por esta razón por la cual *Stuxnet* cambió el paradigma de la seguridad en los sistemas SCADA y ayudó a desmitificar muchas ideas de seguridad de los sistemas aislados previamente mencionadas.

En este escenario, donde ya no son individuos u organizaciones las encargadas de este desarrollo de *malware* con fin de obtener beneficios económicos, sino son Estados financiando el desarrollo de estos *malware* sofisticados para su uso en guerra cibernética, se cambia radicalmente el escenario geopolítico. Según el doctor Olaf Theiler, *Stuxnet* es prueba de que este tipo de *malwares* pone en peligro la integridad física de las personas [14] y la seguridad de las naciones.

Pero *Stuxnet* no es el único *malware* capaz de afectar la producción de una planta industrial, *Flame* es un *malware* predecesor de *Stuxnet* con el propósito de robar información de las operaciones de las empresas, según se explica en el artículo [13].

El problema que genera el espionaje industrial ocasionado por *malwares* como el caso de *Flame*, es la posibilidad de espiar de manera discreta, sin necesidad de estar presente para robar información crucial de las operaciones de una industria o sus secretos; secretos que implican una gran inversión en tiempo y dinero para las empresas, que al ser robados por un tercero, se tienen consecuencias incalculables, como dan cuenta las pérdidas en ventas y el denominado "homicidio corporativo" establecido por Dan McGahan, CEO de AMSC (American Superconductor) luego de que la empresa china Sinovel Windpower robara los datos de una turbina eólica desarrollada por AMSC, ocasionando el colapso de sus ventas en mil millones de dólares [15]. Además, cabe añadir que en Alemania se estima que este espionaje industrial puede causar pérdidas anuales alrededor 50 mil millones de euros [16].

Por esta razón una industria que quiera realizar su transformación digital, debe tener muy presente que se debe invertir en ciberseguridad, pues los ciberataques al ser una amenaza que evoluciona y es cada vez más sofisticada, se genera la necesidad de buscar maneras para proteger los sistemas de estas amenazas. Además de esto, se busca aprovechar el estatus de Colombia

dentro de la industria 4.0, puesto que al tener el primer centro para la cuarta revolución industrial en la región [17], se tienen más oportunidades dadas por el Gobierno para la innovación en esta área.

En Colombia empresas grandes, como el caso de Kaeser Compresores, ya hace uso de los beneficios de la industria 4.0 [18]. Sin embargo, aún falta que más empresas hagan parte de la cuarta revolución industrial, es decir, los procesos de producción de estas no han sido adaptados para potenciar sus funcionalidades haciendo uso de herramientas y equipos propios de la Industria 4.0. Por esta razón, los mecanismos de seguridad desarrollados para estas empresas sólo contemplan los riesgos durante funcionamientos de producción normal (como los indicados en la guía GEMMA [19]).

Adicional a lo anterior, el presente trabajo presenta un interés académico adicional en cuanto busca integrar tendencias tecnológicas basadas en *Blockchain* y técnicas de inteligencia artificial para generar soluciones de aplicación industrial. De esta forma, a través del desarrollo con base en la línea de investigación en Inteligencia Computacional del Grupo de Estudio y Desarrollo en Robótica de la Facultad de Ingeniería Electrónica de la Universidad Santo Tomás, se pretende generar nuevas posibilidades e intereses de trabajo.

Este trabajo busca brindar una solución a la problemática anteriormente expuesta mediante el desarrollo de una técnica de ciberseguridad aplicable al escenario de migración tecnológica hacia la industria 4.0 que afronta Colombia, en el que permita proteger y mantener la integridad de los datos, además de detectar posibles ciberataques.

1.4. Impacto social

En el mundo globalizado que se vive actualmente, las empresas pueden aprovechar las ventajas que trae el Internet de las cosas con el fin de mejorar su producción, además de cumplir con altos estándares internacionales para expandirse dentro del mercado nacional e internacional. Con este panorama llegan oportunidades, y a su vez, vulnerabilidades para dichas empresas [20].

Algunas de las vulnerabilidades son el robo y corrupción de la información, lo cual afecta la integridad de las empresas al no tener asegurados sus datos. Con la técnica implementada en este proyecto, se contribuye a la mitigación del impacto de los ataques por dos razones:

1. Al encriptar la información con una técnica basada en *Blockchain*, esta solo puede ser desencriptada haciendo uso de las llaves que se encuentran en el servidor y en la máquina del cliente, permitiendo que, en caso de ser robaba, la información sea prácticamente inservible.

2. Con la inteligencia artificial que detecta anomalías en la comunicación se logra alertar sobre un posible ataque, para que los equipos de tecnologías de la información de las empresas tomen las medidas correspondientes para reducir el impacto del ataque a tiempo.

De esta forma, se pueden evitar pérdidas económicas y daños en las plantas de las empresas que están realizando la transición hacia la Industria 4.0.

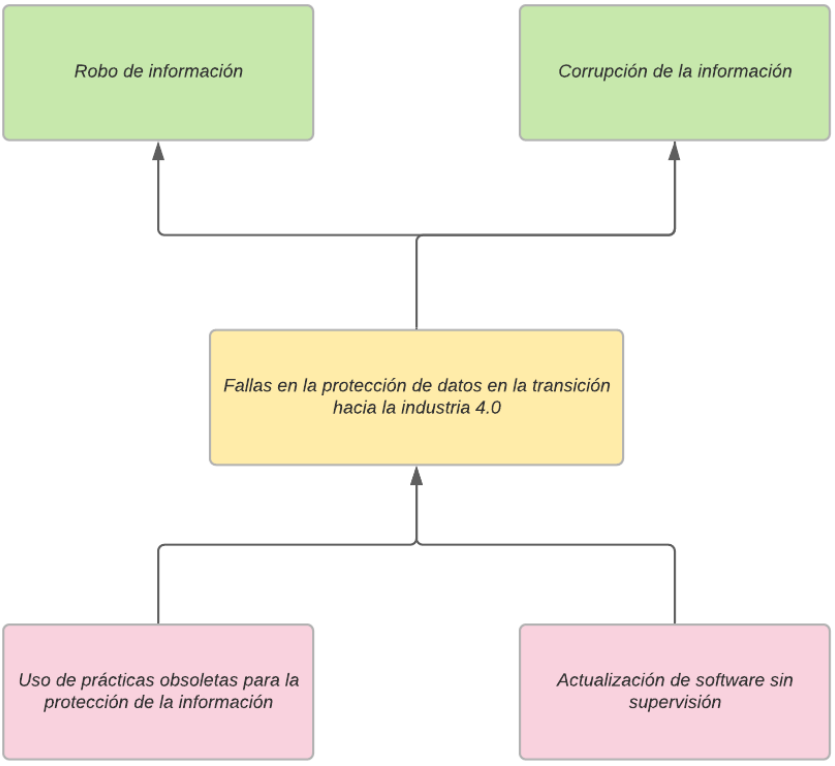

FIGURA 1: Árbol de problema (2021). Autoría propia.
Rojo: Causas.
Amarillo: Problema.
Verde: Consecuencias.

1.5. Objetivos

1.5.1. Objetivo General

Realizar la virtualización de un mecanismo de ciberseguridad basado en *Blockchain* y técnicas de inteligencia artificial para la encriptación y detección de anomalías en la comunicación entre un PLC y la nube para la transición tecnológica hacia la industria 4.0.

1.5.2. Objetivos Específios

- Identificar riesgos y situaciones de ciberataques con base en un conjunto representativo de anomalías en el contexto de la transición tecnológica hacia la Industria 4.0.
- Desarrollar e implementar estrategias de mitigación a ciberataques con base en *Blockchain* y técnicas de inteligencia artificial.
- Verificar con base en los lineamientos de *CYBER SECURITY ASSESSMENTS OF INDUSTRIAL CONTROL SYSTEMS* [21] la mitigación a ciberataques de las estrategias implementadas.

Capítulo 2

Estado del arte

En los últimos años, la industria 4.0 se ha convertido poco a poco en un tema de gran interés para las empresas productoras que pretenden actualizar sus procesos de producción para así seguir vigentes en el mercado. Es por esto que varios autores han tratado este tema desde diferentes aspectos como infraestructura y seguridad, incluyendo los ámbitos de la ciberseguridad e inteligencia artificial. Dentro de estos, se han propuesto diferentes técnicas para proteger los datos de las fábricas frente a un ciberataque a la planta.

2.1. Ciberseguridad

Antes del descubrimiento de *Stuxnet*, se tenía la creencia de que los sistemas de control no podían sufrir un ataque de *malware* ya que, al estar aislados de los otros sistemas, estos no se verían afectados. Pero, con la tendencia de emplear tecnologías de control que funcionan en los sistemas operativos *Windows* y *Linux*, en *TCP/IP* o sistemas de redes, se ha ocasionado el cambio radical de la percepción de ciberseguridad en los sistemas de control, y por ende, reconocer las nuevas amenazas a las que se expone una planta industrial

Otras ideas relacionadas con la seguridad en los sistemas de control son:

- Los sistemas de control están a salvo de ciberataques si no se encuentran conectados a Internet.
- Los detalles técnicos de los sistemas de control solo los conocen los expertos y el personal de campo que no piensan en realizar ciberataques.

- Solo se utilizan dispositivos de almacenamientos nuevos para para el intercambio de datos.
- Los ciberataques se caracterizan por la reducción del rendimiento del equipo y la interrupción de sus tareas.

Al surgir *Stuxnet*, se demostró que estas ideas no son adecuadas para los tiempos actuales, además de dejar al descubierto que los sistemas de control son más vulnerables de lo que se creía. Esto ocasionó que se dejara de lado la idea de que no existen agentes que afecten a los sistemas de control, volviendo necesario replantear y tomar conciencia de la ciberseguridad en los sistemas de control. Debido a que los sistemas de control trabajan en tiempo real, es necesario que se protejan bajo el principio de defensa profunda, es decir, que permita tener mecanismos de protección a ataques redundantes. También cabe resaltar que *Stuxnet* al manejar vulnerabilidades de tipo *Zero-day exploit* (vulnerabilidades del sistema que son desconocidas generalmente por la gente y por el fabricante, los cuales al no ser reparados son usados para aprovecharse de los sistemas), provoca la necesidad de monitorear de manera continua las anomalías que presenten los sistemas de control y analizar sus causas [10].

Cabe resaltar que en el aspecto de seguridad dentro de la industria 4.0 también es importante la resiliencia del sistema, es decir, que el sistema "aprende" a partir de las fallas que ha presentado. En el artículo [22] se realiza el análisis del potencial que tiene la aplicación de tecnología de *Blockchain* en la industria automotriz haciendo énfasis en sus valores de ciberseguridad. También se describen los casos de uso más relevantes, ya que el área de aplicación de *Blockchain* es bastante grande y compleja. En dicho análisis se tienen en cuenta las fortalezas, debilidades y oportunidades para hacer recomendaciones con el fin de guiar a investigadores y compañías en los desarrollos de la industria automotriz ciber-resiliente [22].

2.2. Blockchain

Los investigadores del documento [23] discuten cómo dentro de *Blockchain* caben las aplicaciones de la industria en la manufactura. Además, proponen un enfoque de *middleware* para utilizar los servicios de *Blockchain* y sus capacidades para permitir aplicaciones de manufactura inteligente más seguras, confiables, relevantes y autónomas [23].

Con el trabajo desarrollado en el documento [24] que aplican *Blockchain* para la comunicación entre usuarios finales y sistemas ciber físicos, con el fin de solucionar los problemas de escalabilidad, seguridad y *big data* para las pequeñas y medianas empresas manufactureras.

La propuesta de los investigadores del documento [25] es un sistema seguridad para IoT basado en *Blockchain*, debido a la naturaleza de *Blockchain* que es descentralizada e inmutable, volviendo el sistema más robusto e inmune a los SPF (*Single Point of Failure*, punto único de fallo).

En la investigación realizada en el documento [26], los autores proponen un sistema de *Blockchain* basado en un protocolo de consenso distribuido a través de SDIIoT(*Software-Defined Industrial Internet of Things*, Redes Definidas por Software para Internet de las cosas industrial) para simplificar y proteger los datos y la sincronización de diferentes controladores SDN (*Software-defined Networking*, Redes Definidas por Software), definiéndolo como un proceso de decisión de Markov para optimizar el sistema de manera conjunta definiendo el espacio de estados, las acciones de espacio y las funciones de recompensa.

Un caso que está más ligado a la problemática tratada en el presente proyecto de grado, donde los autores muestran cómo organizar la interacción económica entre agentes utilizando redes de igual a igual basadas en tecnología *Blockchain* no centralizada y contratos inteligentes. La solución que le dan a la problemática de la encriptación es crear un protocolo implementado en el sistema operativo del robot - ROS (entorno de trabajo flexible, con una amplia variedad de herramientas, librerías y paquetes que busca la creación de software complejo para tener robots robustos y con un comportamiento variado [27]) y además presentando *Ethereum Blockchain* en la forma de un software universal para diferentes agentes. El resultado de aplicar esto en varios proyectos es un proyecto de negocio con vehículos aéreos y añadido a esto, un proyecto de educación llamado *smart city* [28].

2.3. Inteligencia Artificial

Los autores del documento [29], plantean la implementación de una red neuronal entrenada para la detección y clasificación de ciberataques en la que, si descubre la intención de un ciberataque, pasa a activar las políticas de seguridad correspondientes. Esta está compuesta por dos fases: análisis de las propiedades de los paquetes y reducción de dimensión de los datos.

También proponen el uso de técnicas como PCA (*Principal Component Analysis*, análisis de componentes principales), para la reducción de los *datasets*. Para el entrenamiento lo describen en tres pasos:

1. Proceso de pre aprendizaje: a partir de un GRBM (*Gaussian Binary Restricted Boltzmann Machine*, máquina de Boltzman Gausiana Binaria Restringida) que transforma los datos reales en código binario para las capas ocultas de la neurona.

2. Proceso de aprendizaje con *Deep Learning*: en este paso se realiza el proceso de aprendizaje donde se reajustan los pesos de las neuronas de la red neuronal, cada proceso es realizado a entre dos capas sucesivas de en las capas ocultas a través de RBM (*Restricted Boltzmann Machine*, máquina de Boltzman Binaria Restringida).

3. Entrenamiento fuera de línea y detección en linea de ciberataques: compuesta por dos fases: pre entrenamiento y ajuste. En pre entrenamiento se recolectan datos sin etiquetar (datos en los que se etiquetan características, propiedades o clasificaciones), mientras que en la fase de ajuste si se hace uso de datos etiquetados.

Después del entrenamiento la neurona ya puede ser utilizada para la detección de paquetes maliciosos en línea.

La investigación desarrollada en el documento [[30], pp. 32765–32782], donde plantean el uso de un sistema de *machine learning* para la detección de DGA (*Domain Generation Algorithm*, Algoritmo para Generacion de Dominios), con un *blacklist* dinámico para una clasificación de DGA o actividad normal más eficiente.

En el documento [31] los autores proponen un *framework* llamado DFEL (*Deep Feature Embedding Learning*), diseñado para prevenir ciberataques con daños irreversibles, detectando intrusiones en entornos de IoT. Explicando la necesidad de usar detectores de posibles ataques en tiempo real. Debido al uso de enormes *dataset* en el entrenamiento de las redes neuronales, plantean el uso de *Deep Feature Embedding Learning*, permitiendo reducir el tamaño de los datos de los *dataset* y la reducción de los tiempos de entrenamiento.

Capítulo 3

Marco Teórico

3.1. Escenario virtualizado

Para crear un escenario virtualizado de industria 4.0 es necesario tener en cuenta qué es esta industria. El término de industria 4.0 surge en Alemania en el año 2011, la cual hace referencia a una política económica gubernamental basada en estrategias de alta tecnología; caracterizada por la automatización, la digitalización de los procesos y el uso de las tecnologías de la electrónica y de la información en la manufactura. Igualmente, por la personalización de la producción, la prestación de servicios y la creación de negocios de valor agregado. Además, también se caracteriza por las capacidades de interacción y el intercambio de información entre humanos y máquinas [32].

Dentro de los elementos que componen un escenario de la industria 4.0 está el PLC (controlador lógico programable). Este es un procesador digital secuencial programable que posee una unidad operativa y una unidad de control programable, el cual actúa sobre las variables de salida mediante la ejecución de una secuencia de instrucciones [33]. Para hacer uso de un PLC en un entorno virtualizado, se hace uso del programa de PLCsim, el cual es un *software* de simulación de PLC perteneciente al *software* TIA PORTAL del fabricante de PLC Siemens, permitiendo realizar conexiones a través de *TCP/IP* [34] entro los distintos programas de desarrollo propio.

TCP/IP es un protocolo de comunicación sobre el cual funcionan la mayoría de las aplicaciones de red, como servidores *web*, correo electrónico, sistemas de transferencia de archivos, entre otros. Su arquitectura está formada por cuatro capas: capa de acceso a la red, capa de red, capa de transporte y capa de aplicación [35].

Para realizar una comunicación entre un dispositivo *A* y un dispositivo *B* en el presente proyecto, se realiza por medio del protocolo *TCP/IP* con una arquitectura cliente - servidor. Este modelo consiste en indicar un intercambio de información, en el que una de las partes debe iniciar el diálogo (cliente) mientras que la otra debe estar indefinidamente preparada para recibir peticiones de establecimiento de dicho diálogo (servidor). Cada vez que un usuario cliente desee entablar un diálogo, primero se deberá contactar con el servidor, enviar una petición y, posteriormente, esperar la respuesta [35]. Esta comunicación se puede realizar usando el lenguaje de programación Python, el cual es un lenguaje de programación de propósito general en alto nivel con baste uso en distintas áreas del conocimiento, diseñado para expresar conceptos en pocas lineas de código, además de contar con una amplia variedad de librerías de código abierto [36].

Para el desarrollo del presente proyecto, por parte del servidor, es necesario instalar y configurar INetSim, que es un *software* de código abierto basado en *Linux* que permite simular un proveedor de servicios de Internet [37]. También, se debe configurar *Wireshark*; este es un programa de código abierto ampliamente usado en los campos de redes, análisis de seguridad, desarrollo y educación, además, permite capturar, visualizar, y analizar los paquetes de datos de una red [38]. Con estos dos *softwares* instalados en el dispositivo de servidor, es posible emular una conexión a un proveedor de Internet y así poder analizar la comunicación que se efectuaría entre el escenario de la nube y el de la planta industrial.

Para implementar los escenarios de la nube y la planta industrial es necesario implementarlas en máquinas virtuales, ambas se ejecutan dentro del programa VirtualBox, un *software* de código abierto que permite la virtualización de distintos sistemas operativos en un mismo entorno. Este programa es desarrollado por Oracle [39].

3.2. Inteligencia artificial

La inteligencia artificial es una disciplina relacionada con la teoría de la computación, cuyo objetivo es emular algunas de las facultades intelectuales humanas en sistemas artificiales. Sus aplicaciones más habituales son el tratamiento de datos y la identificación de sistemas. El diseño de un sistema de inteligencia artificial normalmente requiere del uso de herramientas de varias disciplinas como el cálculo numérico, la estadística, la informática, el procesado de señales, el control automático, la robótica y la neurociencia [40]. El *machine learning* es la evolución de los algoritmos computacionales, con la característica de ejecutar un modelo matemático que asemeja el proceso de aprendizaje humano con el fin de resolver modelos [41]. El aprendizaje

profundo o *deep learning* es un campo del *machine learning*, cuyo objetivo consiste en que las máquinas aprendan a partir del entrenamiento y la observación en un entorno establecido [42].

Para el desarrollo del presente proyecto es necesario hacer uso de técnicas de inteligencia artificial, como un modelo de *machine learning* para la identificación de anomalías. Este modelo hace uso de una red neuronal, la cual es un sistema de procesamiento de información inspirado en la estructura y el funcionamiento de los sistemas neuronales biológicos. Su importancia reside en la capacidad que tiene para aproximar funciones basándose en el conocimiento adquirido a partir del patrón de decisiones desde un conjunto de datos[43]. Adicionalmente se debe establecer dicho conjunto de datos o *dataset*, que corresponde a los contenidos de una única tabla de base de datos o una única matriz de datos de estadística, donde cada columna de la tabla representa una variable en particular, y cada fila representa a un miembro determinado del conjunto de datos que se están tratando. En este se tienen todos los valores que puede tener cada una de las variables, como por ejemplo la altura y el peso de un objeto, que corresponden a cada miembro del conjunto de datos. Cada uno de estos valores se conoce con el nombre de dato. El conjunto de datos puede incluir datos para uno o más miembros en función de su número de filas [44]. En el caso del presente proyecto, este *dataset* está conformado por la información que se analice en la comunicación entre la nube y el entorno industrial.

Para la implementación del modelo de *machine learning* se hace uso del lenguaje de programación python junto con las librerías tensorflow y keras. TensorFlow es una librería que utiliza el álgebra computacional para obtener un compilado de técnicas de optimización, con el fin de lograr cálculos matemáticos más rápidos. Esta fue desarrollada por el equipo de *Machine Learning Inteligence* y *Brain Team* de Google [45]. Por su parte, keras es una librería que permite la construcción modular de modelos de *deep learning* haciendo uso de la librería de tensorFlow [46]. De esta manera es posible crear, diseñar, experimentar, evaluar e implementar un modelo de *machine learning* que se ajuste a las necesidades del presente proyecto.

Teniendo en cuenta que el tipo de datos que se obtienen del *dataset* son cadenas de texto, es necesario hacer uso de técnicas basadas en el procesamiento de lenguaje natural. El procesamiento de lenguaje natural o NLP por sus siglas en inglés, es el área de las ciencias de la computación que investiga como los computadores son capaces de entender y manipular el lenguaje natural en forma de texto o audio [47]. Una arquitectura adecuada para este tipo de tareas es el modelo de las redes neuronales recurrentes son un tipo de red neuronal los cuales contienen conectores cíclicos, lo que les permite ser adecuadas para modelos con datos de entrada secuenciales como es el caso del modelado de lenguaje [48]. En estos casos se aconseja comenzar con una arquitectura *Long Short-Term Memory* o LSTM, contiene unas celadas de memoria las cuales tienen una serie de conexiones que permite tener un almacenamiento temporal del estado de la red,

adicionalmente cuenta con unas celdas multiplicativas llamadas *gates* que tienen como función el control del flujo de la información [48].

3.3. Ciberseguridad

El crecimiento del ciberespacio e Internet ha obligado al Estado y las empresas a desarrollar acciones necesarias que garanticen condiciones mínimas de seguridad para que toda la población pueda utilizarla de forma confiable. La ciberseguridad es el conjunto de técnicas y políticas que establecen medidas que permiten a las personas adquirir las habilidades y competencias necesarias para el uso pleno de Internet [49]. Estas técnicas de cibersugridad son claves al momento de plantear una estrategia de mitigación a los ciberataques.

Una de estas estrategias es la codificación de datos, que consiste en la encriptación de archivos para que no puedan ser descifrados en caso de ser interceptados por un tercero mientras esta información viaja por la red. Solamente a través de un *software* de decodificación conocido el autor de estos documentos encriptados es como se puede volver a codificar la información, por lo que la encriptación informática es simplemente la codificación de la información que se va a enviar a través de la red [50].

Una de las tecnologías que permite la encriptación de los datos es *Blockchain*. Esta hace referencia a una cadena de bloques que contiene transacciones, distribuida de igual a igual. Es criptográficamente segura, inmutable (extremadamente difícil de cambiar) y actualizable solo por consenso o acuerdo entre pares [51]. El protocolo más usado para llegar a estos consensos es *proof of work* o prueba de trabajo, y este es un requisito que deben completar los mineros de *Blockchain* que deseen agregan un nuevo bloque a la cadena de bloques [52].

Otra tecnología clave en *Blockchain* es el protocolo de encriptación asimétrica. Este protocolo utiliza dos claves que pertenecen al emisor del mensaje (pública y privada) para el envío de mensajes. La clave pública se puede entregar a cualquier emisor, mientras que la clave privada únicamente la posee el agente autorizado. El emisor usa la clave pública del destinatario para encriptar el mensaje, y solo la clave privada del receptor podrá desencriptarlo [53].

La asignación de la clave pública se hace a través de certificados digitales –documento electrónico que contiene la clave y la identidad del destinatario y está avalado por una entidad certificadora –, ya que de esta forma se asegura que una determinada clave pertenece a un solo usuario. El proceso que utiliza el protocolo de criptografía asimétrica es el siguiente [53]:

1. Generación de las claves: cada usuario debe contar con sus propias claves. Para esto se utilizan algoritmos de generación de claves públicas y privadas.
2. Asignación de las claves públicas: este proceso se basa en una infraestructura de clave pública (PKI, *Public Key Infrastructure*) donde la identidad del usuario conjuntamente con su clave pública son almacenados en un certificado digital.
3. *Hash* o huella digital: consiste en aplicar la función matemática unidireccional.
4. Firmado del mensaje: encriptación del *hash* del documento con la clave privada del emisor para adjuntarlo al mensaje original.
5. Validación de la integridad: cuando el receptor recibe el mensaje y su firma digital asociada, se debe calcular el *hash* del documento recibido y desencriptar la firma digital con la clave pública del emisor y comparar ambos *hash*, de esta forma se garantiza que el mensaje recibido fue el correcto. En caso de que al comparar los *hash* estos no coincidan, se considera que la información fue alterada.

3.4. Malware

El código malicioso o *malware* se define como un *software* que cumple la intención dañina de un atacante. Este es diseñado con propósitos criminales, políticos y/o maliciosos [54] y cabe resaltar que el daño causado por este ha aumentado drásticamente en los últimos años [55].

Dentro de la amplia variedad de *malwares* y ciberataques que se pueden utilizar, se destacan los siguientes para analizar en este proyecto de grado por su funcionamiento al hacer uso del protocolo *TCP/IP*:

- **Inyección de datos falsos:** La inyección de datos falsos en una categoría de ciberataque consiste en enviar datos o información falsa de la lectura de uno varios instrumentos de medición de una planta, ocasionando que cuando el control actúe, se realicen acciones con base en información que no corresponde a la realidad [56].
- **Malware Andromeda:** Andromeda es un *malware* de los más extendidos en el mundo, el cual utiliza los *e-mail* para propagarse. Al ser un *malware* modular permite tomar capturas de pantalla, identificar teclas pulsadas, entre otras funciones [57].

- **Malware Carberp:** Carberp es un *malware* programado para robar dinero a sus víctimas a partir del espionaje y secuestro en el tráfico de las transferencias bancarias a través de Internet [58].
- **Sniffing:** Es una técnica de análisis de paquetes dentro de las redes de comunicación. En esta, el dispositivo o programa utilizado captura los datos de los paquetes que se envíen o se reciban de un dispositivo a otro, guardando la información para un posterior análisis [59].

Para ejecutar la observación de estas amenazas en la comunicación entre la planta industrial y la nube es necesario realizar un análisis de *malware*. Este consiste en el estudio del comportamiento de los *malwares* y como detectarlos para eliminarlos y bloquearlos. Pueden analizarse ya sea por un análisis dinámico, un análisis estático, análisis del código fuente o análisis de memoria [60]. El análisis de *malware* dinámico es un proceso de ejecución del programa sospechoso en un entorno monitorizado y aislado, con el fin de estudiar el comportamiento del programa [60].

3.5. La nube

Cuando se hace referencia a la nube, se alude a un término que es una forma metafórica de nombrar al Internet. La computación en la nube consiste en los servicios ofrecidos a través de la red tales como correo electrónico, almacenamiento, uso de aplicaciones, etc., los cuales son normalmente accesibles mediante un equipo con conexión a Internet. Al utilizar estos servicios, la información utilizada y almacenada, así como la mayoría de las aplicaciones requeridas, son procesadas y ejecutadas por un servidor en Internet [61].

Un ejemplo de nube es Microsoft Azure, Azure es la plataforma en la nube de Microsoft, permitiendo hospedar aplicaciones y cargas de trabajo existentes en los centros de datos de Microsoft, además de permitir la simplificación de procesos para el desarrollo de nuevas aplicaciones [62].

Capítulo 4

Metodología

Para el desarrollo del proyecto, en primera instancia se realizó una revisión bibliográfica con el fin de identificar los riesgos y amenazas hacia los sistemas de la industria 4.0. Se utilizaron las bases de datos suministradas por la Universidad Santo Tomás tales como IEEE Xplore, Taylor and Francis Online, Science Direct entre otros. Además, se consultaron guías y documentos de carácter normativo elaborados por diferentes autoridades gubernamentales.

4.1. Escenario de pruebas virtualizado

Se realiza el desarrollo y la configuración del escenario de pruebas en un entorno virtualizado. Se usó como base el entorno de pruebas propuesto en el libro Learning Malware Analysis [60]. En este, el autor explica cómo configurar un entorno de pruebas seguro para realizar análisis dinámico de *malwares*. El modelo base de este escenario se ve en la Figura 3.

Este escenario permite ejecutar de una forma segura y aislada de Internet los *malwares* y algoritmos de ciberataques que son de desarrollo propio. Además del modelo base, es necesario instalar y configurar los modelos de la planta industrial y la nube. Para la máquina virtual de la nube se debe configurar y validar un *software* que permita recopilar el flujo de paquetes que reciba la nube y otro *software* que emula una conexión real a la Internet, de cara a engañar a las anomalías que se ejecuten del lado de la planta industrial que traten de realizar una conexión a un *host* en Internet.

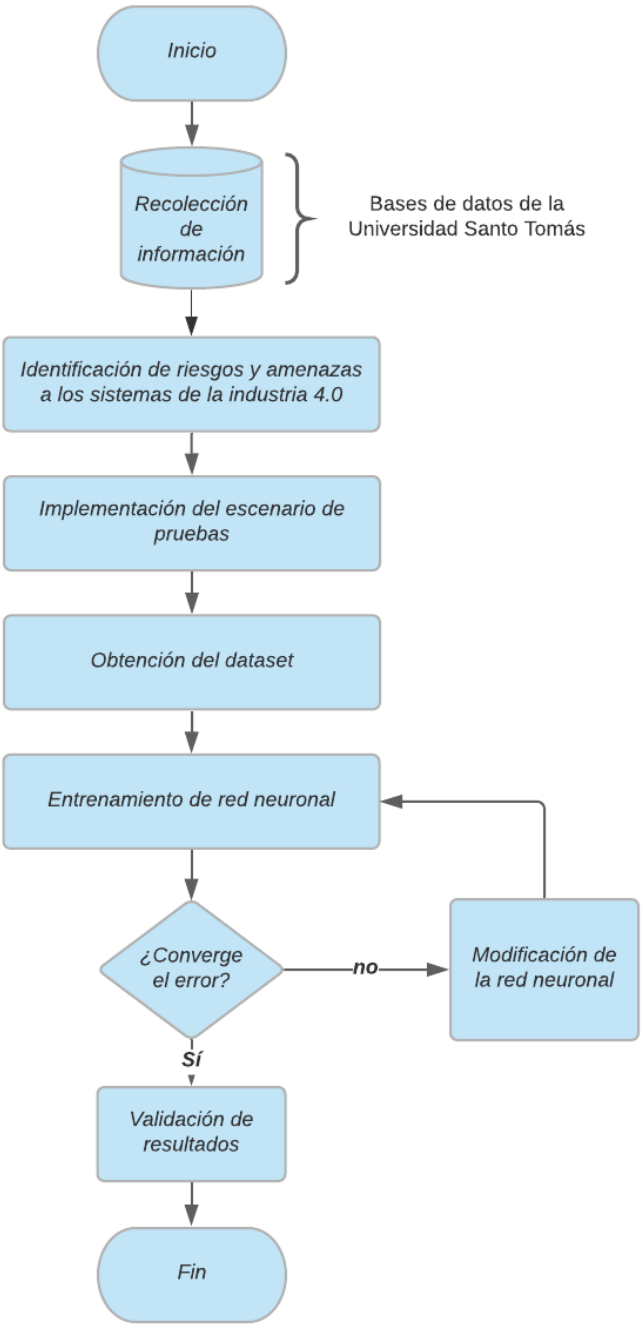

FIGURA 2: Metodología para el desarrollo del proyecto (2021). Autoría propia.

4.2. Identificación del conjunto representativo de anomalías

La identificación del conjunto representativo de anomalías se realizó haciendo uso del repositorio *theZoo - A Live Malware Repository* [63], que contiene muestras vivas de *malwares*. Este

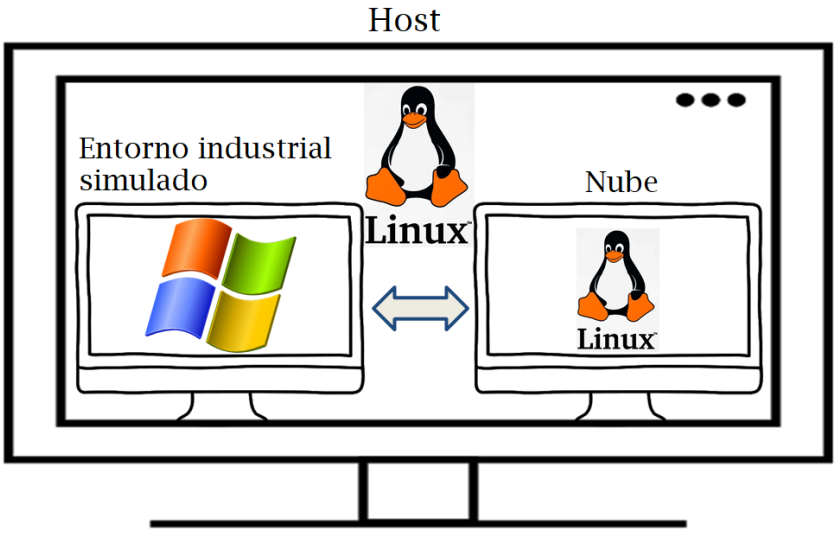

FIGURA 3: Modelo del entorno de pruebas aislado (2021). Autoría propia.

repositorio permite descargar estas muestras vivas con el fin estudiar sus efectos y sus patrones con fines académicos bajo la responsabilidad de cada usuario. A partir de las distintas muestras de *malwares* que se consideró emplear en el entorno industrial en un escenario de infección de *malware* se escogieron:

- Andromeda.
- Carberp.

Estas muestras además de hacer uso del protocolo *TCP/IP*, permiten ser ejecutadas en un entorno virtualizado como el propuesto en este proyecto.

Además de los *malwares*, se consideró hacer uso de escenarios de ciberataques como elementos del conjunto representativo de anomalías. Esto a partir de la necesidad de aumentar las anomalías que permitan realizar la evaluación de la solución. Con los lineamientos dados por CYBER SECURITY ASSESSMENTS OF INDUSTRIAL CONTROL SYSTEMS [21] y el escenario de pruebas, se escogieron los siguientes ciberataques:

- **Ciberataque de sniffing:** Consiste en diseñar un algoritmo que roba los paquetes de la máquina infectada, enviando esta información a un servidor externo haciendo uso del protocolo *TCP/IP*.

- **Ciberataque de data injection:** Consiste en diseñar un algoritmo que envía información falsa al servidor principal de la comunicación entre el escenario industrial y el servidor *host* en la nube.

4.3. Obtención del conjunto de datos

Para obtener el *dataset* con el cual se va a entrenar la red neuronal, se hace uso del programa *Wireshark* que se encarga de registrar los paquetes de datos que pasen a través del puerto de red de la máquina virtual de la nube. Con este *software* se recolectan paquetes de información con sus cabeceras, *IPs* de origen y destino, el protocolo de ese paquete y la longitud de los datos en el lapso de tiempo que se deje ejecutando. Para el *dataset* se establecieron los siguientes tiempos y condiciones de los distintos escenarios para las muestras positivas en el Cuadro 1 y para las muestras negativas en el Cuadro 2.

Obtención de muestras positivas	
Minutos	Programa ejecutado
60	Entorno industrial en condiciones normales

CUADRO 1: Reglas para obtener el *dataset* de muestras positivas.

Obtención de muestras negativas	
Minutos	Programa ejecutado
40	Algoritmo *malware* propio
5	Andromeda
5	Andromeda + Algoritmo *malware* propio
5	Carberp
5	Carberp + Algoritmo *malware* propio

CUADRO 2: Reglas para obtener el *dataset* de muestras negativas.

Después de obtener el *dataset* se debe realizar un proceso de filtrado para eliminar la información irrelevante para la red neuronal. La estructura del *dataset* antes de realizar la limpieza se puede ver en el cuadro 3 y el *dataset* después de eliminar los datos irrelevantes será similar al del cuadro 4.

Ejemplo *dataset* antes de ser preprocesado						
No.	Time	Source	Destination	Protocol	Length	Info
1	0	192.268.1.50	192.268.1.100	TCP	66	57037 >443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK-PERM=1
2	031741	192.268.1.100	192.268.1.50	TCP	66	443 >57037 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK-PERM=1 WS=128
3	279736	192.268.1.50	192.268.1.100	TCP	60	57037 >443 [ACK] Seq=1 Ack=1 WIN=2102272 LEN=0
4	774239	192.268.1.50	192.268.1.100	TLSv1.2	268	Client Hello
5	787412	192.268.1.100	192.268.1.50	TCP	54	443 >57037 [ACK] Seq=1 Ack=215 Win=64128 Len=0

CUADRO 3: Ejemplo *dataset* antes de ser preprocesado.

Ejemplo *dataset* después de ser preprocesado			
Source	Destination	Protocol	Info
192.268.1.50	192.268.1.100	TCP	57037 >443 [SYN] 0 64240 0 1460 256 1
192.268.1.100	192.268.1.50	TCP	443 >57037 [SYN, ACK] 0 1 64240 0 1460 1 128
192.268.1.50	192.268.1.100	TCP	57037 >443 [ACK] 1 1 2102272 0
192.268.1.100	192.268.1.50	TCP	443 >57037 [ACK] 1 215 64128 0

CUADRO 4: Ejemplo *dataset* después de ser preprocesado.

4.4. Consideraciones para el modelo de inteligencia artificial

Teniendo el *dataset* con las muestras positivas y negativas procesadas, se procede a establecer la red neuronal teniendo en cuenta el tipo de datos que se tienen en el *dataset*, razón por la cual se optó por un modelo basado en el procesamiento de lenguaje natural (NLP). En este proyecto se utilizó una red neuronal recurrente (RNN) con una arquitectura *Long Short-Term Memory* (LSTM). Esta arquitectura se elige debido a las diversas aplicaciones en el campo del procesamiento de texto [64].

Posterior a la implementación y entrenamiento de la red neuronal se procede a establecer el escenario de pruebas que permita poner a prueba el modelo. Esto con el fin de establecer si cumple con los objetivos de este proyecto. El escenario debe recibir un archivo en formato *.CSV*

con los registros del monitoreo de puertos que entrega *wireshark* en condiciones normales y anormales.

Capítulo 5

Diseño y procedimiento

5.1. Implementación del escenario de prueba

Por motivos de seguridad y dada la naturaleza del proyecto el cual requiere de la ejecución de *malwares*, es necesaria la implementación de un escenario aislado de la red que permita simular un entorno de industria 4.0. Para esto, se aplicó un escenario basado en el libro *Learning Malware Analysis* [60], en donde se propone un entorno de pruebas para realizar un análisis dinámico de las muestras de *malware*. Este escenario está compuesto por una máquina física (computador) con sistema operativo *Linux*, el cual contiene dos máquinas virtuales:

1. Máquina virtual con sistema operativo *Windows*, la cual contiene los programas que simulan una planta industrial y los escenarios de ciberataques.
2. Máquina virtual con sistema operativo *Linux*, la cual contiene los programas que simulan una conexión a Internet, Wireshark para analizar el trafico de red y los servidores de *Host-planta* y *Host-malware*.

El escenario completo puede verse en funcionamiento en la Figura 4, donde se ven las dos máquinas virtuales y la máquina *Host*.

Con el fin de tener el escenario aislado de la red de Internet, se crea una red interna para las máquinas virtuales con la configuración del adaptador de red en *Host*, la cual permite que entre las dos máquinas pueda existir un canal de comunicación sin conectarse a una red externa. Para esto, se configuran las direcciones *IP* de las máquinas de la siguiente forma:

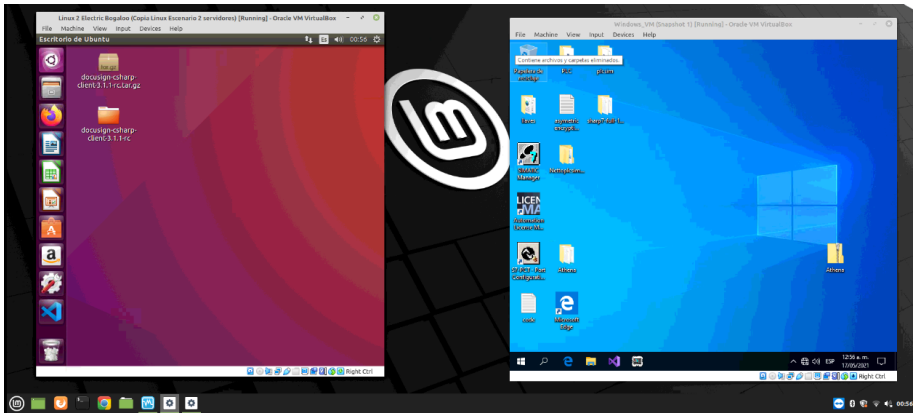

FIGURA 4: Escenario aislado implementado (2021). Autoría propia.

- *IP server*: 192.168.1.100
- *IP cliente (planta industrial)*: 192.168.1.50

Con las direcciones *IP* y la red interna de las máquinas virtuales configuradas, se termina de ajustar la máquina virtual *Linux* con el fin de que tenga el comportamiento de una conexión a Internet, además de ejecutar los dos servidores *Host*. Esto se puede validar contrastando la configuración *IP* de las máquinas virtuales con la Figura 5 para la máquina virtual *Linux* y Figura 6 para la máquina virtual *Windows*. En esta última se puede observar su *IP* propia y la puerta de enlace para la máquina virtual, siendo esta la *IP* de la máquina virtual a la que está conectada (máquina virtual con *Linux*).

```
miguel@miguel-VirtualBox: ~
miguel@miguel-VirtualBox:~$ ifconfig
enp0s3    Link encap:Ethernet  direcciónHW 08:00:27:16:3d:20
          Direc. inet:192.168.1.100  Difus.:192.168.1.255  Másc:255.255.255.0
          Dirección inet6: fe80::a00:27ff:fe16:3d20/64 Alcance:Enlace
          ACTIVO DIFUSIÓN FUNCIONANDO MULTICAST  MTU:1500  Métrica:1
          Paquetes RX:6686 errores:0 perdidos:0 overruns:0 frame:0
          Paquetes TX:6492 errores:0 perdidos:0 overruns:0 carrier:0
          colisiones:0 long.colaTX:1000
          Bytes RX:675897 (675.8 KB)  TX bytes:1429578 (1.4 MB)

lo        Link encap:Bucle local
          Direc. inet:127.0.0.1  Másc:255.0.0.0
          Dirección inet6: ::1/128 Alcance:Anfitrión
          ACTIVO BUCLE FUNCIONANDO  MTU:65536  Métrica:1
          Paquetes RX:508 errores:0 perdidos:0 overruns:0 frame:0
          Paquetes TX:508 errores:0 perdidos:0 overruns:0 carrier:0
          colisiones:0 long.colaTX:1000
          Bytes RX:43722 (43.7 KB)  TX bytes:43722 (43.7 KB)
```

FIGURA 5: Configuración de dirección *IP* del servidor (2021). Autoría propia.

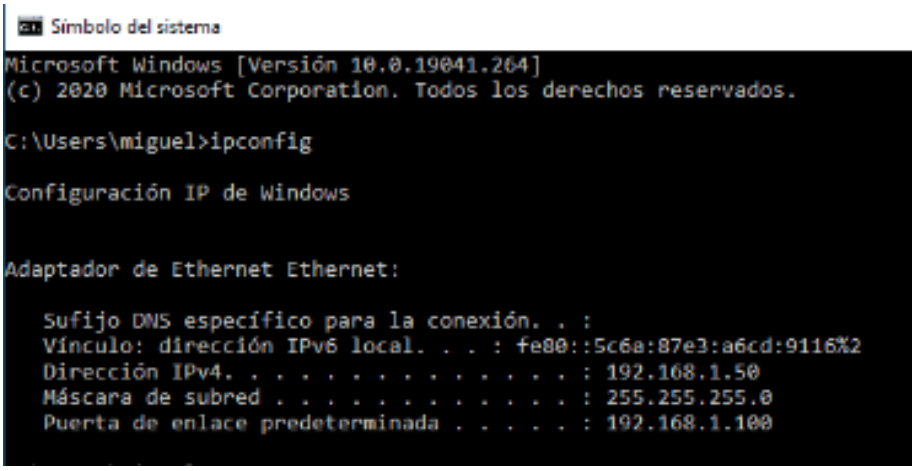

FIGURA 6: Configuración de dirección *IP* de la planta industrial (2021). Autoría propia.

5.2. Diseño de la planta industrial

En la máquina virtual *Windows* que simula una planta industrial, se implementó un algoritmo que permite emular el comportamiento de un motor trifásico, como se muestra en el algoritmo 1.

En este algoritmo se realiza la conexión con el servidor y un *LocalHost*. Este último permite establecer la conexión con la interfaz del PLC del programa *PLCsim*. Al establecerse esta conexión se procede a emular el PLC y controlarlo con el programa que ejecuta el algoritmo previamente mencionado. Para captar la información del *PLCsim*, se crea y lee el *buffer* que recibe todos los datos entre el algoritmo y la interfaz del PLC. Adicionalmente en este escenario industrial se emula por software un variador de frecuencia conectado al PLC, el cual manejará los cambios de velocidad que el operador desee sobre un motor trifásico.

Las interfaces del *PLCsim* y del algoritmo de desarrollo propio se pueden apreciar en la figura 7 y figura 8. En la interfaz del *PLCsim* se puede ver y controlar el estado del motor (en marcha o detenido) además de la velocidad.

Además de tener la configuración del entorno industrial, en la maquina virtual *Windows* es necesario tener los algoritmos que emulen una situación de ciberataque y las muestras de *malware* vivas (Andromeda y Carberp), que al ejecutarse pondrán en amenaza al entorno industrial. Estos algoritmos de ciberataque están representados en los algoritmos 2 para el ciberataque de *sniffing* y 3 para el ciberataque *data injection*.

Algorithm 1: Industrial plant

```
ip;
if connection = true then
    print(çonnected to server");
else
    print("not connected to server");
end
while cicle=true do
    buffer;
    read buffer;
    PLC variables;
    if engine is running then
        if selected speed then
            change speed;
        else
            do not cange speed;
        end
    else
        engine is stopped;
    end
    write on buffer;
end
```

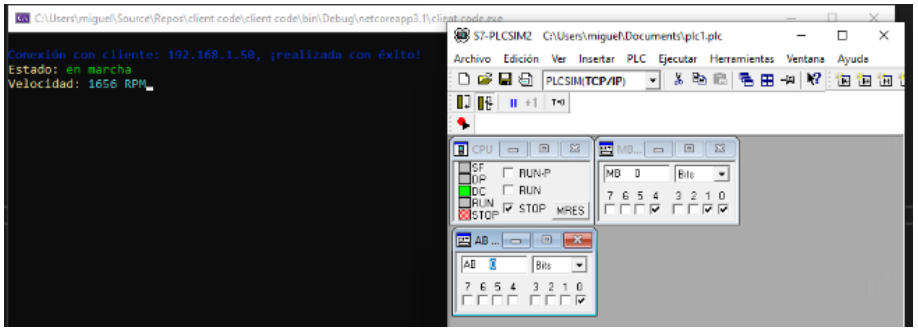

FIGURA 7: Interfaz de PLC con motor en marcha (2021). Autoría propia.

El algoritmo de *sniffing* se encarga de recopilar los meta datos de los paquetes que pasan por la comunicación de la máquina infectada. Este *malware* envía esta información a un *host* haciendo uso del protocolo *TCP/IP*.

El segundo algoritmo propuesto, correspondiente al ciberataque de *data injection*, se encarga de enviar datos a *host-planta*. Estos datos falsos que no corresponden con los de la planta industrial, ocasionan que el sistema de *host-planta* utilice datos basura, además de entorpecer la comunicación entre la planta industrial y *host-planta*.

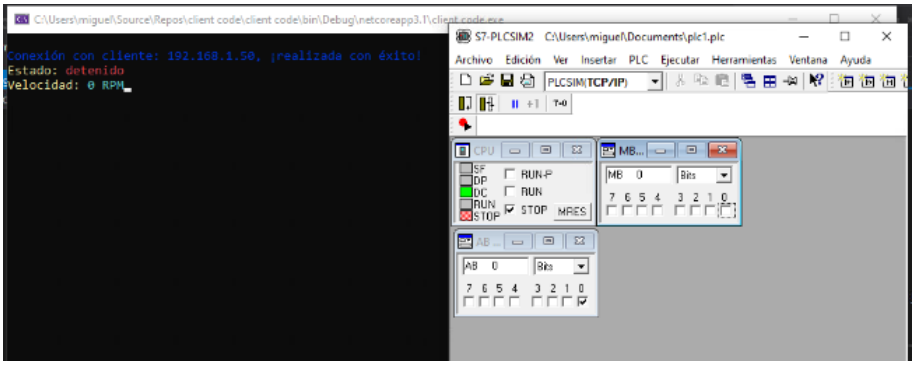

FIGURA 8: Interfaz de PLC con motor detenido (2021). Autoría propia.

Algorithm 2: Malware sniffing

Result: Simulate a sniffing attack
Malware Host Server Connection;
if *connection = true* **then**
    while *cicle=true* **do**
        Read data packets from infected PC;
        Message = data packets;
        Send *Message* to Host Server;
        Wait 5 seconds;
    end
else
    print(.Error: not connected to server");
end

Algorithm 3: Malware data injection

Result: Simulate a data injection attack
Industrial Plant Host Connection;
if *connection = true* **then**
    print(çonnected to server");
    while *cicle=true* **do**
        Message = Create false data;
        Send *Message* to Host Server;
        Wait 5 seconds;
    end
else
    print(.Error: not connected to server");
end

5.3. Diseño de la nube

Con el fin de tener un entorno que se asemeje a la nube de un entorno de industria 4.0 y estar aislado de la red de Internet para realizar el análisis dinámico de *malware*, se configuró una maquina virtual con sistema operativo *Linux*, que se conecta por una red interna del programa *VirtualBox* a la máquina virtual con *Windows*.

Además de configurar la red interna de las máquinas virtuales y realizar una validación con un *Ping* entre ambas máquinas virtuales, en la máquina de *Linux* se debe configurar *INetSim* y Wireshark. *INetSim* es el software que permite emular una conexión a la Internet de cara a la máquina virtual *Windows*; mientras que Wireshark es el software encargado de capturar y registrar todos los paquetes de datos que pasen a través de la máquina virtual de *Linux*, siendo este registro de paquetes los datos que alimentarán al modelo de inteligencia artificial.

Dentro de los algoritmos propios que se necesitan ejecutar en la máquina virtual *Linux* es necesario contar con *Host-planta* y *Host-malware* los cuales recibirán los mensajes provenientes de los clientes, en este caso la comunicación de la planta industrial, los *malware* cuando se ejecuten (*malware data injection* y *malware sniffing*). Los algoritmos que representan estos *Host* se pueden ver en el algoritmo 4 para la planta industrial y en el algoritmo 5 para el *malware sniffing*.

Algorithm 4: Host-planta

```
Host-planta IP config;
while true do
    if Recive new message then
        Decrypt new message;
        if error then
            print(.Error: cant decrypt message!!");
        else
            print(new message);
        end
    else
        Wait until recive a new message;
    end
end
```

5.4. Preprocesamiento del dataset

A pesar de que Wireshark entrega los listados de los paquetes de datos capturados en la máquina virtual *Linux*, se obtienen datos como se observa en el Cuadro 3. Parte de las columnas

Algorithm 5: Host-malware

```
Host-malware IP config;
while true do
    if Recive new message then
        print(new message);
    else
        Wait until recive a new message;
    end
end
```

se deben eliminar para que los datos queden estructurados como se observa en el Cuadro 4.

Esto se logra con el algoritmo 6, el cual no solo está diseñado para eliminar las columnas que no se van a utilizar, sino que también permite eliminar los paquetes de datos de la columna *Info* que se encuentren repetidos. Adicionalmente, se considera que no todos los paquetes tienen los mismos parámetros en su información. Por consiguiente, los parámetros faltantes se deben asignar con un valor de cero para después ser organizados por orden alfabético y finalmente eliminar el nombre de los mismos. De esta manera, la información es completamente numérica (a excepción de la forma en que se envió el paquete) y se consigue que esté ordenada para poder obtener un patrón, permitiendo que la red pueda identificarlo y clasificar correctamente los paquetes de la comunicación entre el *PLC* y la planta emulada.

Con este preprocesamiento se puede eliminar la redundancia al entrenar, ya que es posible garantizar que no existen datos repetidos y que tampoco existan los mismos datos entre clases que ocasione un conflicto en la red neuronal al momento de realizar el entrenamiento.

Otro aspecto a considerar es si el *dataset* es balanceado, ya que si no se tienen cantidades similares de cada tipo de muestra, el entrenamiento no será bueno ya que tendría la tendencia a clasificar todas las muestras como una sola clase.

5.5. Modelo inteligencia artificial

5.5.1. Red neuronal

Para seleccionar el tipo de red neuronal primero se partió del tipo de problema que se tiene para el presente proyecto, el cual es de identificación. Por lo tanto, se escogió una red neuronal *LSTM*, la cual es una extensión de las redes neuronales recurrentes, que básicamente amplía

Algorithm 6: Filter Dataset

Normal Conditions = Read normal condition CSV file;
Anormal Conditions = Read anormal conditions CSV files;
while *each packet in Normal Conditions* **do**
    Delete repeated packet data in *Normal Conditions*;
    Add missing data in *Normal Conditions*;
    Organize data in *Normal Conditions*;
    Delete letters in information of *Normal Conditions*;
end
while *each packet in Anormal Conditions* **do**
    Delete repeated packet data in *Normal Conditions*;
    Delete repeated packet data in *Anormal Conditions*;
    Add missing data in *Anormal Conditions*;
    Organize data in *Anormal Conditions*;
    Delete letters in information of *Anormal Conditions*;
end
Verify repeated data between *Normal Conditions* and *Anormal Conditions*
if *Normal Conditions and Anormal Conditions don't have repeated data* **then**
    Write new CSV files for *Normal Conditions* and *Anormal Conditions* including only *Source*, *Destiny*, *Protocol*, *Info* columns.
else
    print(.Error: Can't create CSV files");
end

su memoria para aprender de experiencias importantes que ocurrieron hace mucho tiempo durante el entrenamiento [48].

La arquitectura de la red *LSTM* se observa en la figura 9, donde se puede apreciar que tiene una celda para memoria en color naranja. Esta celda se destaca por retener su valor durante un tiempo largo, permitiendo que recuerde los valores importantes y no solamente los últimos obtenidos.

5.5.2. Entrenamiento de la red neuronal

El entrenamiento de la red neuronal tiene dos partes importantes: el ajuste de los datos y el modelo del entrenamiento.

1. Ajuste de los datos: este se realiza con la finalidad de codificarlos y ajustarlos para que la información de cada paquete tenga la misma longitud. Este ajuste se realizó utilizando *one hot representation*, la cual toma cada lista dentro de una tupla y la codifica. De esta

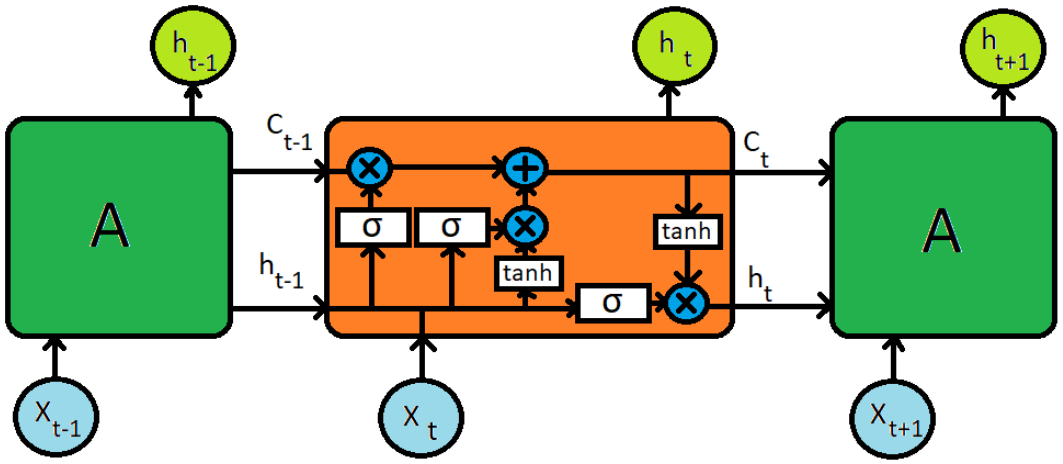

FIGURA 9: Modelo LSTM (2021). Autoría propia.

manera, cada lista logra tener la misma longitud. Ya teniendo esto, en la información hay algunos paquetes con mayor longitud que otros, razón por la cual se hace *padding* con ceros.

2. Entrenamiento de la red: este necesita de ciertos parámetros para poder ejecutarse. Para este proyecto, se seleccionaron los siguientes y sus valores:

   - Función de activación: *sigmoid*
   - Función de pérdida: *binary crossentropy*
   - Optimizador: adam
   - Métrica: *accuracy*
   - Épocas: 40
   - *Batch size:* 100

5.6. Validaciones técnicas del escenario virtualizado

A partir del escenario de pruebas que se propuso con el fin de implementar el mecanismo de ciberseguridad virtualizado, se realizó una serie de pruebas que permita validar si satisfacen las necesidades, las cuales son:

- Transmisión de información del entorno industrial alojado en la máquina virtual *Windows* al servidor *Host* alojado en la máquina virtual *Linux* haciendo uso de la tecnología del cifrado asimétrico.

- Validar en el *Host* del entorno industrial de la máquina virtual de *Linux* las alertas de errores al tratar de descifrar los mensajes recibidos en un ataque de data injection y seguir operando en condiciones normales.
- Ejecutar múltiples ciberataques que hagan uso del protocolo *TCP/IP* al entorno industrial, con el fin de estudiar los paquetes de datos que viajen a través de este protocolo e identificar si existe algún ataque al entorno industrial analizando la comunicación del *Host* y el entorno industrial.

5.7. Validación del proyecto

El último objetivo planteado de este proyecto de grado es verificar las estrategias de mitigación de ciberataques en base a los lineamientos del documento CYBER SECURITY ASSESSMENTS OF INDUSTRIAL CONTROL SYSTEMS [21], dentro de las consideraciones que se deben para la evaluación se debe:

1. Tener presente hasta qué punto se conoce el funcionamiento interno de todo el *hardware* y *software* de la infraestructura. En este caso se conocen los algoritmos de desarrollo propio. Sobre los demás *softwares* como Wireshark, INetSim, PLCsim, se desconoce su funcionamiento interno, estando en un punto intermedio del conocimiento total del funcionamiento de la planta industrial.
2. Los objetivos de seguridad que priorizan la ciberseguridad en escenarios industriales ponen en primer lugar la disponibilidad de los datos. La integridad y la confidencialidad de los datos ocupan los puestos dos y tres en la escala de prioridades, respectivamente.
3. El documento explica los cinco pasos para realizar la evaluación los cuales son:
    a) **Planeación:** En esta etapa se identificaron las vulnerabilidades a evaluar, definiendo unas metas.
    b) **Evaluación:** Se establecen las pruebas de seguridad que se van a realizar en la evaluación de la vulnerabilidad. En esta etapa se definen los vectores de ataque y se realiza un escaneo de los puertos y sus vulnerabilidades.
    c) **Reporte:** Resultados de la evaluación realizada.
    d) **Corrección de las vulnerabilidades:** Recomendaciones y sugerencias a realizar con el fin de mejorar la seguridad del entorno industrial.

e) **Pruebas de validación:** A partir de las recomendaciones se vuelve a realizar la evaluación con el fin de validar que se han corregido o mitigado las vulnerabilidades.

Para validar este proyecto y considerando las limitaciones de un entorno industrial virtualizado, se tomó la decisión de ejecutar los pasos *a)*, *b)* y *c)*. Los motivos radican en que en estos pasos es posible tener una información clara del sistema a partir de las consideraciones que hace; los puntos *d)* y *e)* están pensados en reforzar las fallas del sistema. Este refuerzo del sistema no se encuentra contemplado los alcances de este proyecto.

Dentro de la guía se hace mención de las distintas vulnerabilidades que se deben considerar en un entorno industrial. Aunque la gran mayoría están enfocadas a un entorno más completo y utilizando otros sistemas que no hacen parte de una planta industrial, si hacen parte de la infraestructura de las organizaciones. Tomando las vulnerabilidades que se ajustan al escenario virtualizado de este proyecto, se optó por evaluar las vulnerabilidades en la *web* y la manipulación e inyección de datos y comandos.

Adicionalmente proponen el siguiente diagrama de flujo que sirve de modelo para el desarrollo del informe de evaluación como se puede ver en la figura 10. En este diagrama se pueden observar los elementos clave que permiten evaluar de manera correcta cada escenario con las vulnerabilidades que se consideren necesarias.

Como complemento a la evaluación, existen tres métricas de evaluación de la infraestructura las cuales son:

- **Métricas base**
  - Vector de acceso.
  - Complejidad de acceso.
  - Autenticación.
  - Impacto de confidencialidad.
  - Impacto de integridad.
  - Impacto de disponibilidad.
- **Métricas temporal**
  - Explotabilidad.
  - Nivel de corrección.
  - Reporte de confidencia.

- **Métricas del ambiente**
  - Potencial daño colateral.
  - Distribución de objetivos.
  - Requisito de confidencialidad.
  - Requisito de integridad.
  - Requisito de disponibilidad.

Con estas métricas y el documento modelo para escoger dependiendo del nivel (alto bajo o medio) y su correspondiente descripción, se puede complementar la evaluación haciendo un análisis detallado en los tres grupos de métricas principales.

FIGURA 10: Diagrama de flujo reporte assessment (2021). Autoría propia.

Capítulo 6

Resultados

6.1. Riesgos y situaciones identificadas del conjunto de anomalías

Dentro de los riesgos y situaciones que se pueden destacar del conjunto representativo de anomalías se logra identificar cómo funcionan los *malwares* estudiados, y de estos se sabe que:

- Dentro del catálogo de *malwares* del repositorio de *theZoo - A Live Malware Repository*, no muchos hacen uso del protocolo *TCP/IP*. Aunque se puedan utilizar para tener una amplia variedad de escenarios para analizar, si no utilizan el puerto de comunicaciones no se pueden detectar si llegan a infectar la planta industrial, ya que los alcances de este proyecto se limitan a monitorizar la comunicación de la planta industrial con la nube para la detección de las anomalías.
- Dentro en cada archivo que contiene las muestras vivas de *malware* es posible encontrar la documentación de funcionamiento y los ejecutables de una manera sencilla; en otros casos, se vuelve más difícil ejecutar las muestras al estar entre carpetas con los códigos fuentes, dificultando el proceso de ejecución del *malware*.
- El uso de algoritmos propios para emular los ciberataques de *data injection* y *sniffing* pese a no ser un ataque real, podemos detectar la actividad que generan por medio de los paquetes del protocolo de *TCP/IP*.

Adicionalmente, en un principio se consideró emplear 5 y no 2 muestras de *malware*. El motivo de este cambio, a pesar de que todas las muestras utilizan el protocolo de *TCP/IP* se produjo por:

1. Las dificultades en la ejecución de las muestras.
2. Algunas muestras no eran adecuadas para ejecutar en el entorno virtualizado planteado, esto debido a que se requería como máquina *host* y planta industrial con **Windows** en ambas máquinas, dificultando así su operación con la configuración planteada.

Finalmente, a partir del conjunto mencionado se lograron identificar las siguientes malas prácticas que exponen una planta a un ciberataque:

- Uso de contraseñas no seguras en los equipos de trabajo.
- Uso de unidades de almacenamiento externo.
- Conexión de los equipos a Internet sin utilizar *softwares* de protección como antivirus y *firewall*.

6.2. Resultados del Escenario Virtual

Validando la infraestructura propuesta para el presente proyecto de grado, se realizaron distintas pruebas que permiten determinar si realmente se da cumplimiento a las necesidades planteadas en el capítulo cinco.

Los resultados para la comunicación entre el entorno en condiciones normales y el servidor host-planta se puede observar en la Figura 11, la cual permite validar una correcta comunicación y descifrado de la información subministrada por el entorno industrial.

En la Figura 12 se aprecia cómo el servidor host-planta informa de un error al momento de descifrar los datos que recibe del ataque de *data injection*, adicionalmente de seguir recibiendo la información proveniente del entorno industrial.

Verificando el escenario bajo múltiples ciberataques, se puede ver en la Figura 13 cómo el servidor host-planta, además de recibir el ataque de *data injection*(como se observó en la Figura 12), se observa un segundo servidor. Este segundo *Host* corresponde al ataque de *sniffing* host-malware, recibiendo los paquetes de datos que espía del entorno industrial.

Dentro de los últimos resultados relacionados al entorno virtualizado, se destaca el funcionamiento de todos los elementos necesarios para emular el entorno industrial, los ciberataques que utilicen el protocolo de *TCP/IP*, y el programa *Wireshark* que permite analizar la comunicación entre la máquina virtual *Windows* (entorno industrial) y la máquina virtual *Linux* (la

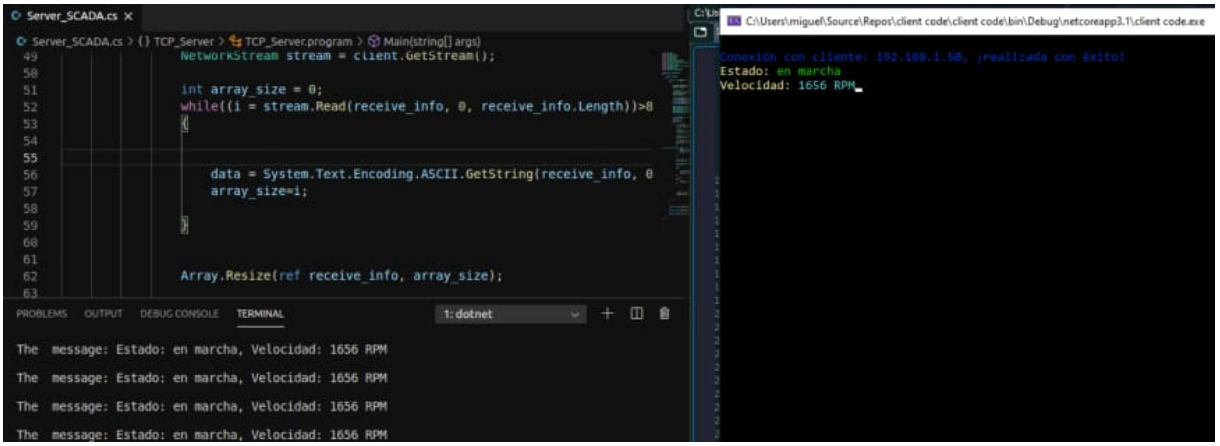

FIGURA 11: Entorno industrial virtualizado en condiciones normales (2021). Autoría propia.

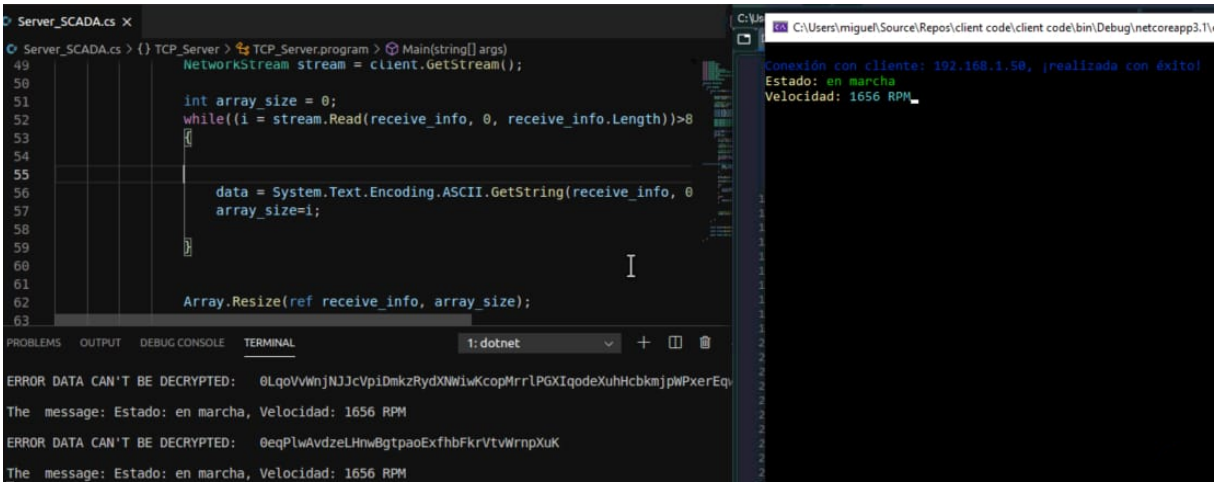

FIGURA 12: Entorno industrial virtualizado con ciberataque de data injection (2021). Autoría propia.

nube). Como se observa en la Figura 14 se aprecia como todos los paquetes de datos del escenario planteado en la Figura 12 son capturados por el programa *Wireshark*.

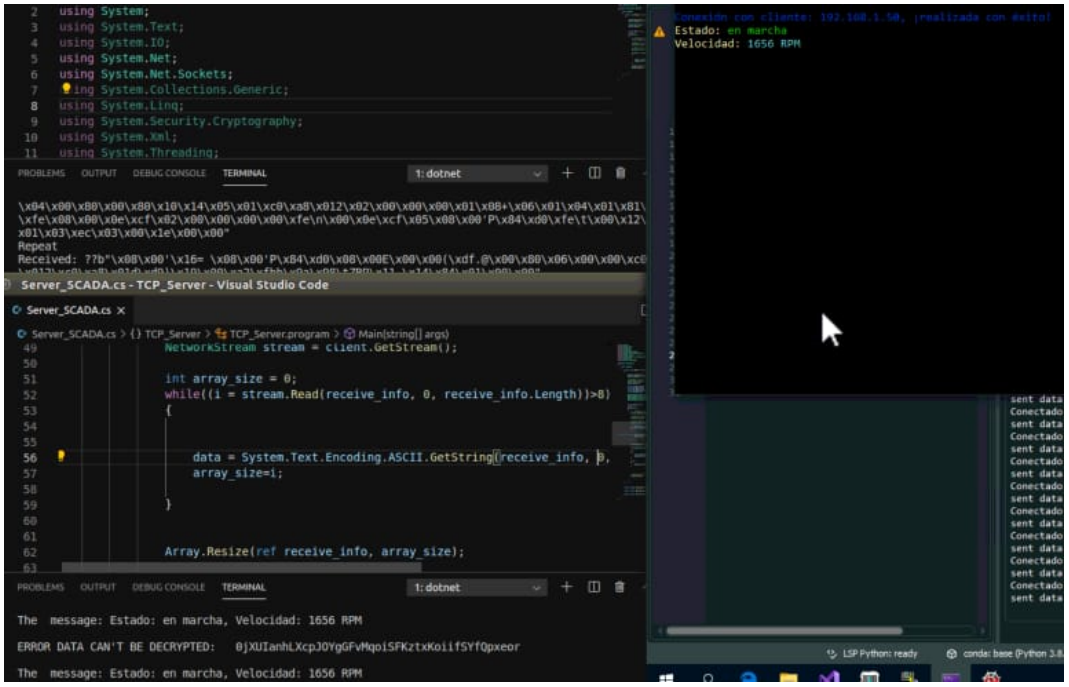

FIGURA 13: Entorno industrial virtualizado con ciberataque de data injection y sniffing (2021). Autoría propia.

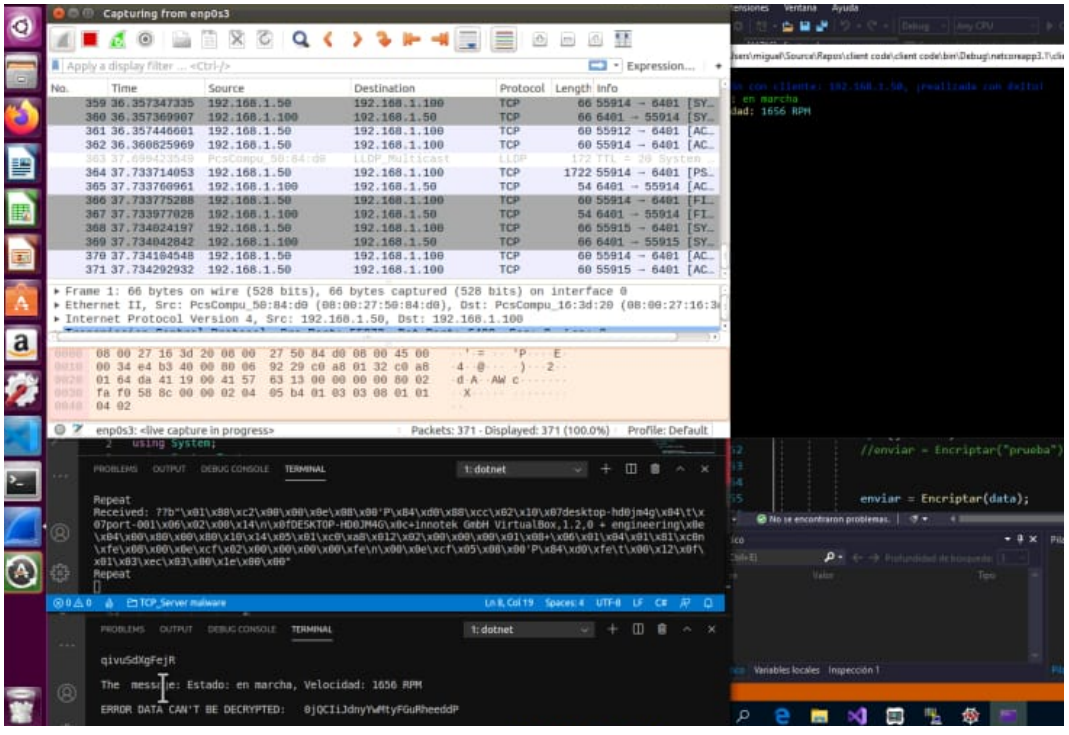

FIGURA 14: Entorno industrial virtualizado con captura de datos bajo un ciberataque de data injection y sniffing (2021). Autoría propia.

6.3. Resultados red neuronal LSTM

Debido a que el problema de este proyecto es de clasificación, el resultado del entrenamiento se valida con una matriz de confusión. Esta indica que tan bien se clasificaron los datos de un porcentaje del *dataset* que fue separado previamente para verificación. En la figura 15 se puede apreciar que los valores de la diagonal principal son mayores que el resto de los valores de la matriz, lo cual indica que está clasificando la mayoría de muestras correctamente.

Adicionalmente, también se observa que el valor AUC, el cual corresponde al de una métrica que indica la probabilidad de que un dato sea clasificado correctamente [41], es elevado.

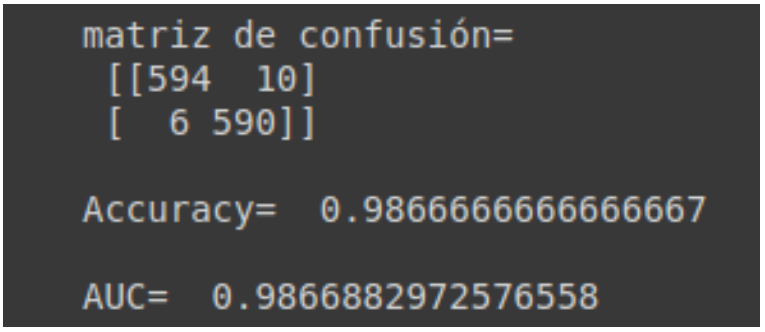

FIGURA 15: Resultado del entrenamiento (2021). Autoría propia.

Para tener una validación adicional, se probó el modelo de la red neuronal con un archivo distinto al del entrenamiento. En la figura 16, se observa el resultado de esta prueba.

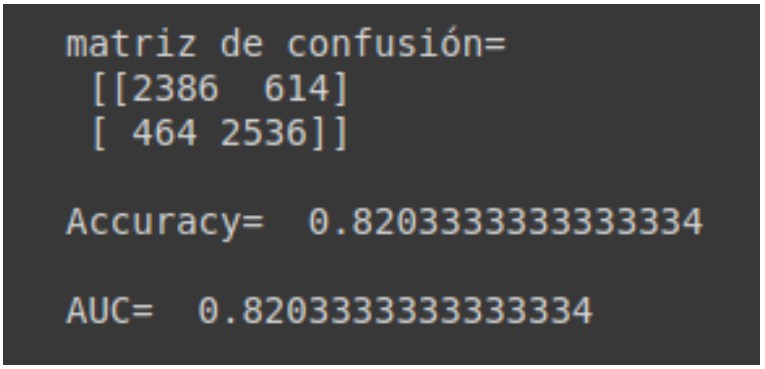

FIGURA 16: Resultado de la prueba (2021). Autoría propia.

6.4. Resultados Blockchain

Dentro de los intentos para la implementación completa de la tecnología de *Blockchain* se obtuvieron los siguientes resultados:

- A partir del algoritmo de *Blockchain* en python propuesto por IBM [65], se realizaron los primeros test de la tecnología. A pesar de permitir modificar el código fuente, se encontraron varias dificultades técnicas al momento de realizar las modificaciones necesarias a puntos tales como:
  - Reemplazar el algoritmo PoW (*Proof of Work*, prueba de trabajo) que consume muchos recursos de computo y tiempo por otro más rápido y eficiente en computo.
  - Reemplazar el modelo de web chat original del programa por un modelo cliente servidor.
- Como alternativa al algoritmo previamente mencionado, se tomó la decisión de buscar una plataforma de nube pública (AWS, IBM cloud) que permitiera una mayor flexibilidad. Después de hacer uso de los créditos de prueba de ambas plataformas, no fue posible tener una aplicación funcional para el desarrollo del presente proyecto.

A pesar de que no fue posible la implementación completa de *Blockchain*, se estudiaron las distintas tecnologías que utiliza *Blockchain* para operar [66], tomando la decisión de implementar la encriptación asimétrica con el fin de dar cumplimiento a los objetivos planteados en el presente proyecto. Haciendo uso de la encriptación asimétrica se logró:

1. Tener una mayor libertad a la hora de escribir código fuente propio en distintos lenguajes de programación.
2. Obtener en menor tiempo los datos de la planta industrial, sin consumir grandes cantidades de computo.
3. Lograr diferenciar un mensaje auténtico de la planta industrial frente a los demás mensajes que no utilicen las llaves de cifrado usadas en la encriptación asimétrica.

6.5. Validación del proyecto

Los resultados de esta sección se encuentran en el anexo 1.

Capítulo 7

Epílogo

7.1. Conclusiones

A partir de los resultados descritos en el capítulo 6 se destacan las siguientes conclusiones:

- Es posible identificar los patrones de ciberataques analizando los paquetes de la comunicación de la planta industrial y la nube. Sin embargo, a pesar de poder detectar estas anomalías en la comunicación que haga uso del protocolo *TCP/IP*, si se llegase a presentar un ciberataque que no utilice ese protocolo de comunicación, o que se ejecute de manera local sin hacer uso de ninguna comunicación, este modelo planteado se vuelve insuficiente para detectar la anomalía. Es por esa razón que se aconseja emplear distintos análisis de seguridad en la infraestructura de la planta industrial adicionales al análisis del PLC y la nube en un escenario de industria 4.0.
- El modelo de red neuronal propuesto en el presente proyecto de grado entrega métricas de precisión del *82.03 %*, permitiendo que este modelo de *machine learning* sea capaz de diferenciar de manera correcta los paquetes de datos pertenecientes a un escenario de condiciones normales y de un ciberataque aproximadamente el *82.03 %* del total de los paquetes analizados en por el modelo.
- El uso de la encriptación asimétrica entre la planta industrial y la nube permite proteger los datos de una manera sencilla y rápida. Al poder ser implementado en distintos lenguajes de programación y sistemas operativos, permitiendo la interoperabilidad de los distintos componentes de la planta industrial se anexen y requieran de una comunicación con la nube, sin poner en riesgo el correcto funcionamiento del modelo planteado.

- A partir de los resultados de la evaluación del sistema que se realizó en base de los parámetros planteados en el documento CYBER SECURITY ASSESSMENTS OF INDUSTRIAL CONTROL SYSTEMS [21], podemos destacar como a pesar de los riesgos de seguridad presentes en el escenario virtualizado, el uso de encriptación asimétrica permite reforzar la seguridad sin poner en riesgo la disponibilidad y la integridad de la información.

7.2. Trabajos Futuros

Con base en el trabajo presentado en este documento, se hace énfasis en los siguientes puntos que permitirán desarrollar nuevos proyectos para las personas que opten por complementar el proyecto de este documento.

- Automatizar los procesos de captación, procesamiento y análisis de los paquetes de datos que se encuentran monitorizando desde la planta industrial.
- La implementación de este proyecto no en un escenario virtualizado, sino haciendo uso de en una planta física, con equipos e instrumentos usados en la industria, con el fin de realizarlo en condiciones físicas.
- A pesar de encontrar dificultades en la implementación completa de la tecnología de *Blockchain*, se recomienda no desestimar esta tecnología haciendo uso de otros métodos que no se llegaron a contemplar para este proyecto.
- Con el fin de vincular este proyecto con tecnologías empleadas en la industria, se aconseja validar los modelos y técnicas de inteligencia artificial que ofrecen nubes públicas como Microsoft Azure haciendo uso de servicios como Azure Machine Learning, que permite hacer uso de una plataforma especializada en modelos de *machine learning* de desarrollo propio o con los modelos sugeridos por Microsoft, el propósito de esta recomendación es emplear una inteligencia artificial que se ajuste mejor al escenario propuesto.
- Adicionalmente se sugiere la posibilidad de integrar el proyecto con las tecnologías SIEM, la idea de esta sugerencia es poder integrar la información ante una amenaza que obtenemos de este proyecto con la tecnología SIEM, esta tecnología permite recopilar metadatos de distintas fuentes con el fin de centralizar y automatizar tareas en caso de ciberataques.

Bibliografía

[1] C. Rameback. «Process automation systems-history and future». En: *EFTA 2003. 2003 IEEE Conference on Emerging Technologies and Factory Automation. Proceedings (Cat. No.03TH8696)*. Vol. 1. 2003, 3-4 vol.1. DOI: 10.1109/ETFA.2003.1247680.

[2] Ludovic Noirie, Michel le pallec y Nesrine Ammar. «Towards automated IoT service recommendation». En: mar. de 2017, págs. 103-106. DOI: 10.1109/ICIN.2017.7899397.

[3] Nour Moustafa y col. «A new threat intelligence scheme for safeguarding industry 4.0 systems». En: *IEEE Access* 6 (2018), págs. 32910-32924.

[4] «Cybersecurity for Industry 4.0». En: (2018), pág. 8.

[5] *¿Quién crea malware y por qué?* https://encyclopedia.kaspersky.es/knowledge/who-creates-malware-and-why/. (Visitado 2019).

[6] Portafolio. «Hay empresas que pierden hasta $4.000 millones por ciberataques». En: *Portafolio* (2019).

[7] Revista Dinero. «4 de cada 10 empresas en América Latina sufrieron ciberataques en los últimos años». En: *Revista Dinero* (2019).

[8] *¿No hay víctimas pequeñas para los cibercriminales*. https://www.kaspersky.es/about/press-releases/2017_no-small-victims-for-cybercriminals/. (Visitado 2019).

[9] MINTIC. *SEGURIDAD Y PRIVACIDAD DE LA INFORMACIÓN*. URL: https://www.mintic.gov.co/gestionti/615/articles-5482_Modelo_de_Seguridad_Privacidad.pdf.

[10] Toshio Miyachi y col. «Myth and reality on control system security revealed by Stuxnet». En: *SICE Annual Conference 2011*. IEEE. 2011, págs. 1537-1540.

[11] IWONDER BBC. «El virus que tomó control de mil máquinas y les ordenó autodestruirse». En: *El virus que tomó control de mil máquinas y les ordenó autodestruirse* (2019).

[12] The citizen. «Ciberarmas; en el centro de la actualidad». En: *Ciberarmas; en el centro de la actualidad* (2018).

[13] David Kushner. «The real story of stuxnet». En: *ieee Spectrum* 50.3 (2013), págs. 48-53.

[14] Olaf Theiler. «Nuevas amenazas: el ciberespacio». En: *Revista digital de la OTAN* (2011).

[15] IWONDER BBC. «Las enormes dimensiones del espionaje industrial de China (y cómo contribuyó a la guerra comercial con Estados Unidos)». En: *Las enormes dimensiones del espionaje industrial de China (y cómo contribuyó a la guerra comercial con Estados Unidos)* (2018).

[16] Risi Weber. «Espionaje industrial y económico – nuevas formas de ataque». En: *Espionaje industrial y económico – nuevas formas de ataque* (2010).

[17] S Constain. «Colombia en la cuarta revolución industrial». En: *Computerworld* (2019).

[18] Lopez. «¿Está cerca la Industria 4.0 en Colombia?» En: *La Republica* (2018).

[19] P Ponsa, R Vilanova y M Diaz. «Gemma guide approach for the introduction of the human operator into the automation cycle». En: *IFAC Proceedings Volumes* 39.6 (2006), págs. 285-290.

[20] Ana Inés Basco y col. *Industria 4.0: fabricando el futuro*. Vol. 647. Inter-American Development Bank, 2018.

[21] Lopez. «CYBER SECURITY ASSESSMENTS OF INDUSTRIAL CONTROL SYSTEMS». En: *CYBER SECURITY ASSESSMENTS OF INDUSTRIAL CONTROL SYSTEMS* (2011).

[22] Paula Fraga-Lamas y Tiago M Fernández-Caramés. «A review on blockchain technologies for an advanced and cyber-resilient automotive industry». En: *IEEE Access* 7 (2019), págs. 17578-17598.

[23] Nader Mohamed y Jameela Al-Jaroodi. «Applying blockchain in industry 4.0 applications». En: *2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC)*. IEEE. 2019, págs. 0852-0858.

[24] Ali Vatankhah Barenji y col. «Blockchain-based ubiquitous manufacturing: a secure and reliable cyber-physical system». En: *International Journal of Production Research* 58.7 (2020), págs. 2200-2221.

[25] Rahul Agrawal y col. «Continuous security in IoT using blockchain». En: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2018, págs. 6423-6427.

[26] Chao Qiu y col. «Blockchain-based software-defined industrial Internet of Things: A dueling deep Q-learning approach». En: *IEEE Internet of Things Journal* 6.3 (2019), págs. 4627-4639.

[27] ROS. «About ROS». En: *About ROS* (2017).

[28] Aleksandr Kapitonov y col. «Blockchain based protocol for economical communication in industry 4.0». En: *2018 Crypto valley conference on blockchain technology (CVCBT)*. IEEE. 2018, págs. 41-44.

[29] Khoi Khac Nguyen y col. «Cyberattack detection in mobile cloud computing: A deep learning approach». En: *2018 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE. 2018, págs. 1-6.

[30] Yi Li y col. «A machine learning framework for domain generation algorithm-based malware detection». En: *IEEE Access* 7 (2019), págs. 32765-32782.

[31] Yiyun Zhou y col. «Deep learning approach for cyberattack detection». En: *IEEE INFOCOM 2018-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE. 2018, págs. 262-267.

[32] Carmen Berenicice Ynzunza Cortés, Juan Manuel Izar Landeta y Jacqueline Guadalupe Bocarando Chacón. «El entorno de la industria 4.0: implicaciones y perspectivas futuras». En: *Conciencia tecnológica* 54 (2017), págs. 33-45.

[33] Enrique Mandado Pérez, Jorge Marcos Acevedo y Celso Fernández Silva. *Automatas programables y sistemas de automatizacion/PLC and Automation Systems*. Marcombo, 2009.

[34] Siemens. *SIMATIC S7-PLCSIM Manual del usario*. URL: `https://support.industry.siemens.com/cs/document/109758848/descarga-del-simatic-s7-plcsim-advanced-v2-0-sp1-de-prueba-(trial)?dti=0&lc=es-WW` (visitado 15-01-2021).

[35] Amelia Zafra y col. «Diseño de aplicaciones cliente/servidor para el aprendizaje de las tecnologıas de comunicación». En: *Iniciación a la Investigación* (2013).

[36] KR Srinath. «Python–the fastest growing programming language». En: *International Research Journal of Engineering and Technology* 4.12 (2017), págs. 354-357.

[37] Rycka Septiasari y Yogha Restu Pramadi. «A study on windows-based ransomware implications on linux operating system using compatibility layer wine based on dynamic analysis». En: *IOP Conference Series: Materials Science and Engineering*. Vol. 1007. 1. IOP Publishing. 2020, pág. 012120.

[38] Usha Banerjee, Ashutosh Vashishtha y Mukul Saxena. «Evaluation of the Capabilities of WireShark as a tool for Intrusion Detection». En: *International Journal of computer applications* 6.7 (2010), págs. 1-5.

[39] Pradyumna Dash. *Getting started with oracle vm virtualbox*. Packt Publishing Ltd, 2013.

[40] Raúl Benıtez y col. *Inteligencia artificial avanzada*. Editorial UOC, 2014.

[41] Issam El Naqa y Martin J Murphy. «What is machine learning?» En: *machine learning in radiation oncology*. Springer, 2015, págs. 3-11.

[42] Rayan Alshamrani y Xiaogang Ma. «Deep Learning». En: *Encyclopedia of Big Data*. Ed. por Laurie A. Schintler y Connie L. McNeely. Cham: Springer International Publishing, 2019, págs. 1-5. ISBN: 978-3-319-32001-4. DOI: 10.1007/978-3-319-32001-4_533-1. URL: https://doi.org/10.1007/978-3-319-32001-4_533-1.

[43] Jaime Alberto Villamil Torres y Jesús Alberto Delgado Rivera. «Entrenamiento de una red neuronal multicapa para la tasa de cambio euro-dólar (EUR/USD)». En: *Ingeniería e investigación* 27.3 (2007), págs. 106-117.

[44] J. Quiñonero-Candela y col. «Introduction to Dataset Shift». En: *Dataset Shift in Machine Learning*. 2009, págs. 1-1.

[45] Giancarlo Zaccone. *Getting started with TensorFlow*. Packt Publishing Ltd, 2016.

[46] Nikhil Ketkar. «Introduction to keras». En: *Deep learning with Python*. Springer, 2017, págs. 97-111.

[47] Gobinda G Chowdhury. «Natural language processing». En: *Annual review of information science and technology* 37.1 (2003), págs. 51-89.

[48] Hasim Sak, Andrew W Senior y Françoise Beaufays. «Long short-term memory recurrent neural network architectures for large scale acoustic modeling». En: (2014).

[49] Carolina Sancho. «Ciberseguridad. Presentación del dossier/Cybersecurity. Introduction to Dossier». En: *URVIO. Revista Latinoamericana de Estudios de Seguridad* 20 (2017), págs. 8-15.

[50] Carlos Arturo Carvajal Chávez. «La encriptación de datos empresariales: ventajas y desventajas». En: *RECIMUNDO* 3.2 (2019), págs. 980-997.

[51] Imran Bashir. *Mastering blockchain*. Packt Publishing Ltd, 2017.

[52] P. Rajitha Nair y D. Ramya Dorai. «Evaluation of Performance and Security of Proof of Work and Proof of Stake using Blockchain». En: *2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV)*. 2021, págs. 279-283. DOI: 10.1109/ICICV50876.2021.9388487.

[53] Paola Maritza Velasco Sánchez. «Análisis de los mecanismos de encriptación para la seguridad de la información en redes de comunicaciones». Tesis de mtría. PUCE, 2015.

[54] R. R. Branco y G. N. Barbosa. «Distributed malware analysis scheduling». En: *2011 6th International Conference on Malicious and Unwanted Software*. 2011, págs. 34-41. DOI: 10.1109/MALWARE.2011.6112324.

[55] Andreas Moser, Christopher Kruegel y Engin Kirda. «Limits of static analysis for malware detection». En: *Twenty-Third Annual Computer Security Applications Conference (ACSAC 2007)*. IEEE. 2007, págs. 421-430.

[56] Le Xie, Yilin Mo y Bruno Sinopoli. «False data injection attacks in electricity markets». En: *2010 First IEEE International Conference on Smart Grid Communications*. IEEE. 2010, págs. 226-231.

[57] Omid E David y Nathan S Netanyahu. «Deepsign: Deep learning for automatic malware signature generation and classification». En: *2015 International Joint Conference on Neural Networks (IJCNN)*. IEEE. 2015, págs. 1-8.

[58] Ralph Dolmans y Wouter Katz. «Rp1: Carberp malware analysis». En: (2013).

[59] Felix Fuentes y Dulal C Kar. «Ethereal vs. Tcpdump: a comparative study on packet sniffing tools for educational purpose». En: *Journal of Computing Sciences in Colleges* 20.4 (2005), págs. 169-176.

[60] KA Monnappa. *Learning Malware Analysis: Explore the concepts, tools, and techniques to analyze and investigate Windows malware*. Packt Publishing Ltd, 2018.

[61] O Mejıa. «Computación en la nube». En: *ContactoS* 80 (2011), págs. 45-52.

[62] Microsoft. *Get started guide for Azure developers*. URL: `https://docs.microsoft.com/en-us/azure/guides/developer/azure-developer-guide`.

[63] ytisf. *theZoo aka Malware DB*. URL: `https://thezoo.morirt.com/`.

[64] Alex Sherstinsky. «Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network». En: *Physica D: Nonlinear Phenomena* 404 (2020), pág. 132306.

[65] IBM. *Get started with blockchain - IBM Developer*. URL: `https://developer.ibm.com/technologies/blockchain/tutorials/develop-a-blockchain-application-from-scratch-in-python/` (visitado 21-02-2020).

[66] Hongwen Hui y col. «Survey on Blockchain for Internet of Things.» En: *J. Internet Serv. Inf. Secur.* 9.2 (2019), págs. 1-30.

Anexo 1
Informe de evaluación de ciberseguridad:

Introducción:

Con el fin de dar cumplimiento a los objetivos planteados, se realizó este informe de evaluación de la infraestructura planteada para la planta industrial virtualizada y todos los elementos que la componen.

Los objetivos que se establecieron para la evaluación son:

1. Evaluar el comportamiento de la planta industrial haciendo énfasis en las vulnerabilidades *web* y de manipulación e inyección de datos y comandos.
2. Determinar los impactos de estas vulnerabilidades en la infraestructura planteada.
3. Establecer las mejoras en la mitigación de ciberataques a partir de las estrategias empleadas en el presente proyecto de grado.

Este informe de evaluación es muy importante, ya que nos permite evaluar bajo métricas y reglas diseñadas para entornos industriales reales, esta guía de evaluación fue planteada por el departamento de *Homeland Security* del gobierno de los Estados Unidos. Dando así las garantías de que el presente trabajo se está evaluando con criterios enfocados en la mejora de seguridad de los entornos industriales.

Para realizar el proceso de evaluación se realizará teniendo en cuenta las limitantes que ofrece el escenario virtualizado frente a un entorno real. Adicionalmente se realizará únicamente con vulnerabilidades *web* y de manipulación e inyección de datos y comandos como se planteó en los objetivos de la evaluación.

Evaluación:

Para la evaluación del sistema, las vulnerabilidades a ejecutar son los algoritmos de *data injection* para la vulnerabilidad de manipulación e inyección de datos y comandos, y *sniffing* para la vulnerabilidad *web*. Ambas vulnerabilidades se van a ejecutar en simultáneo, evaluando a partir de las métricas establecidas los puntos débiles y fuertes de la solución.

Métricas de evaluación:

Métricas base	**Valor métrica**	**Descripción**
Vector de acceso	Local	A pesar de que es posible atacar el sistema desde una red

		adyacente, es necesario tener a los agentes ejecutando de manera local.
Complejidad de acceso	Bajo	Siendo un entorno virtualizado de una planta industrial, no se están teniendo consideraciones de acceso de usuarios que permita gestionar el control de acceso a los sistemas. siendo un ítem nulo o bajo.
Autenticación	Ninguno	El sistema no cuenta con sistemas de autenticación que permita dar acceso a los usuarios autorizados.
Impacto en la confidencialidad de los datos	Ninguno	No se cuenta con información confidencial que afecte la operación o la integridad del sistema.
Impacto en la integridad de los datos	Completo	Pese a no estar utilizando un control avanzado en el escenario propuesto, contar con información errónea de los sistemas de medición puede generar serios problemas en los sistemas de control.
Impacto en la disponibilidad de los datos	Completo	A pesar de que en el entorno industrial planteado solo se está realizando el monitoreo de la velocidad de un motor trifásico y su estado, no es crítico la disponibilidad de la información. Se recomienda tener en cuenta evaluar cada variable para sistemas y mediciones críticas.

Métricas temporales	**Valor métrica**	**Descripción**
Explotación de vulnerabilidades	Alto	A pesar de utilizar las vulnerabilidades, el sistema sigue operando y trabajando de manera adecuada, identificando las y ser capaz de informar al usuario.
Nivel de corrección	No definido	No se hace uso de protocolos de comunicación seguros en los programas de comunicación.
Informe de confianza	No definido	Este parámetro no aplica esta evaluación.

Métricas del	**Valor**	**Descripción**

entorno	métrica	
Potencial daño colateral	Medio - alto	Los ataques de inyección de datos pese a no representar una amenaza al poderse identificar, puede saturar el servidor en caso de tomar las medidas adecuadas para este tipo de ataques. Adicionalmente, el *sniffing* es una amenaza para el robo de credenciales de los usuarios autorizados.
Distribución de los objetivos.	Alto	Dada la configuración del escenario de la planta industrial, contamos con el 100% de la infraestructura vulnerable a ciberataques.
Requisitos de seguridad	Medio	El escenario completo a pesar de tener falencias de seguridad, no representan una situación catastrófica que afecte las operaciones normales.

Conclusiones:

- El sistema cuenta con deficiencias en el control de autenticación de usuarios al momento de acceder a los sistemas industriales, permitiendo que un atacante pueda realizar modificaciones en los sistemas.
- La disponibilidad e integridad de los datos son críticos en el sistema, el uso del encriptado asimétrico permite que la comunicación de los sistemas sea rápida y logrando salvaguardar la información que se comunique.
- El sistema es capaz de seguir operando a pesar de estar en ataque de *data injection* y *sniffing*, se recomienda reforzar las tareas de control de puertos de los sistemas, además de reforzar la comunicación con protocolos que refuercen la seguridad de los *Sockets*.
- El sistema de seguridad permite reforzar distintos puntos de la planta industrial como es la disponibilidad e integridad de la información, logrando aumentar los puntos de seguridad del sistema.