Técnica de Machine Learning para la Prevención del Malware-Ransomware

Walter Albeiro Úsuga Bedoya

Director

Mauricio Amariles Camacho

Codirector

Silvana Lorena Vallejo Córdoba

Tecnológico de Antioquia - Institución Universitaria

Ingeniería en Software

Medellín, Colombia.

2021

Dedicatoria

La inmortalidad, producto del miedo al fin, de incertidumbre a un mundo desconocido, el anhelo de todo aquel que desea perdurar en el tiempo. Cuenta un mito que no un hombre, sino un ave a través de la chispa caída de la espada de un Querubín sobre ella, hizo que esta ardiera en llamas... Tres días después el ave de hermoso plumaje rojo resurgió de sus cenizas, y este ciclo empezó a repetirse por toda la eternidad, el Fénix, el símbolo, tal vez, de la esperanza que todo hombre tiene y que jamás debe abandonar su espíritu.

Indudablemente al igual que cualquier hombre subsisto con temor, una reacción normal al tener una vida llena de amor, al haber tenido la fortuna de compartir este camino con compañeros, amigos y familia. Hoy a pesar de tantas dificultades, de tanto esfuerzo y de tanto insistir, soy mi propio Fénix, y decido renacer entre mis propias cenizas, decido con orgullo y con la sabiduría que he acumulado todo este tiempo dedicar este símbolo de mi paso por la vida a mi amada esposa, Sandra Yamile Gutiérrez, la mujer que nunca dudó en tenderme una mano, su apoyo incondicional y sobre todo su cariño que fue un motor para mí. A mis hijos por haberme sabido entender y esperar para verme salir adelante y que esto sea también símbolo de perseverancia y esfuerzo para ellos. A mi madre porque fruto de su crianza y su amor es que decidí empezar este recorrido por la vida y tener lo que tengo hoy en día.

A todos, Dios les pague.

Agradecimientos

A mi Dios ante todo por permitirme llegar hasta el final de esta meta propuesta de hace varios años, agradecimiento especial a mis tutores D. Mauricio Amariles y C. Silvana Lorena Vallejo, quienes han tenido una paciencia infinita, a mi esposa por ayudarme y estar en los momentos más difíciles en los que veía imposible el alcance de esta meta, mis dos hijos porque me guardaron paciencia en los momentos en que ellos me necesitaron y supieron entender mi propósito, a mi madre y mi hermano Jorge que siempre me han apoyado y confiado en mí, aún en los momentos más difíciles.

Resumen

El malware-ransomware es un programa malicioso cuyo objetivo es infectar un sistema para tomar control del mismo, una vez hecho esto, modifica, sustrae, distrae, incomoda y daña parcial o totalmente el software o hardware impidiendo llevar a cabo todos sus procesos y ocasionando pérdidas. Las técnicas usadas actualmente para su detección se están quedando cortas debido a los nuevos paradigmas tecnológicos que aumentan el intercambio de información a través de la red. Personas inescrupulosas crean software malicioso empleando herramientas de alta tecnología, como la Inteligencia Artificial (IA) y vectores sofisticados para engañar a los usuarios; frente a estas nuevas modalidades de virus informático muchas personas y empresas no cuentan con las herramientas, ni protocolos de seguridad suficientes. En este contexto, este trabajo de investigación propone una técnica de detección de malware-ransomware usando la inteligencia artificial, específicamente, técnicas de aprendizaje automático; para lo cual, se efectuó un estudio exploratorio, basado en el método deductivo y el análisis documental de 35 publicaciones de nuevo conocimiento. Los resultados permiten identificar un incremento en la utilización de las técnicas de aprendizaje automático para la detección de diversos malware, destacándose, según la literatura consultada, la aplicación de los modelos de Máquina Vector Soporte (lineal y no lineal) (14%), Naïve Bayes Classifier (13%) y Random Forest (12%); aunque se evidenciaron limitaciones en materia de detección del malware ransomware explícitamente. Es este sentido, se propone una técnica de prevención que es de carácter integradora, porque aprovecha las capacidades de las técnicas empleadas por otros autores y, además, potencia los niveles de precisión y exactitud de estas en cuanto a la detección, la clasificación y la evaluación del malware ransomware.

Palabras clave: aprendizaje automático, ransomware, detección de malware, análisis de malware.

Abstract

Malware-ransomware is a malicious program that infects a system to take control of it, once this is done, it modifies, subtracts, distracts, annoys and partially or totally damages the software or hardware, preventing all its processes from being carried out and causing losses. The techniques currently used for its detection are falling short thanks to the new technological paradigms that increase the exchange of information through the network. Unscrupulous people create malicious software using high-tech tools such as Artificial Intelligence (AI) and sophisticated vectors to deceive users. Faced with these new forms of computer virus, many people and companies do not have the tools or sufficient security protocols. In this context, this research work proposes a malware-ransomware detection technique using artificial intelligence, specifically, machine learning techniques; for which, an exploratory study was carried out, based on deductive method and documentary analysis of 35 new knowledge publications. The results allow identifying an increase in the use of machine learning techniques for different malware forms detection. According to the literature consulted, stand out the application of: Machine Vector Support models (linear and non-linear) (14%), Naïve Bayes Classifier (13%) and Random Forest (12%); although limitations were explicitly evident in ransomware malware detection. In this sense, a prevention technique is proposed that is integrative in nature, because it takes advantage of the capabilities of the techniques used by other authors and, in addition, enhances the levels of precision and accuracy of these in terms of detection, classification and evaluation ransomware malware.

Keywords: Machine-Learning, ransomware, malware detection, malware analysis.

Tabla de Contenido

1. Introducción 13

2. Marco del Proyecto 17

2.1. Definición del Problema 17

2.2. Justificación del Problema 24

2.3. Formulación del Problema 26

3. Marco Contextual 27

3.1. Aspectos Generales. 27

3.2. Antecedentes 29

3.3. Hipótesis 33

3.4. Objetivos 33

3.4.1. Objetivo General 33

3.4.2. Objetivos Específico 34

4. Marco Metodológico 35

4.1. Metodología 35

4.2. Tipo de Investigación 36

4.3. Definición del Alcance 38

5. Marco Teórico – Conceptual 39

5.1. La Ciberseguridad 39

5.2. Malware ransomware 40

5.3. Delincuentes Informáticos 41

5.4. Brecha de Seguridad Informática 41

5.5. Tipos de Malware 42

5.6. Análisis del código fuente de un ransomware escrito en Python 45
5.7. Inteligencia Artificial 49
5.8. Machine Learning 49
5.9. Técnicas de Machine-Learning 50
6. Desarrollo del Proyecto 52
6.1. Identificación de las técnicas de Machine-Learning para la detección de Malware–Ransomware 52
6.1.1. Support Vector Machine (SVM) (lineal y no lineal: kernelizado). 57
6.1.2. Naive Bayes Classifier (NBC) 59
6.1.3. Random Forest (bosque aleatorio) 61
6.1.4. Árbol de decisión, con énfasis en el algoritmo J48 64
6.1.5. Deep Neural Network (DNN) 68
6.1.6. Técnicas de Machine Learning en la literatura aplicadas específicamente en la detección del Malware Ransomware 73
6.2. Caracterización del Malware Ransomware y las principales técnicas que existen para su detección 79
6.3. Construcción de la propuesta de la Técnica de Detección de Malware-Ransomware 92
6.3.1. Fase 1. Selección de los datos. 92
6.3.2. Fase 2. Clasificación de malware 100
6.3.3. Fase 3. Detección del malware ransomware y de evaluación 103
7. Resultados y Discusión 106
8. Impacto Esperado 111
9. Conclusiones 112

10. Recomendaciones Futuras 114

Referencias..... 115

Anexos 126

Índice de Figuras

Figura 1. Eficacia de un antivirus, según la detección de malware y falsos positivos 21

Figura 2. Estadísticas ransomware en usuarios corporativos..... 28

Figura 3. Esquema de funcionamiento ransomware 40

Figura 4. Brecha de seguridad informática 41

Figura 5. Tipos de Malware 42

Figura 6. Análisis del Código Fuente al ejecutarse un Ransomware (parte 1). 46

Figura 7. Análisis del Código Fuente al ejecutarse un Ransomware (parte 2). 47

Figura 8. Análisis del Código Fuente al ejecutarse un Ransomware (parte 3). 48

Figura 9. Frecuencia de uso de las técnicas de Machine Learning identificadas en la literatura . 54

Figura 10. Diagrama de flujo de la fase 1 de la técnica propuesta 100

Figura 11. Diagrama de flujo de la fase 2 de la técnica propuesta 103

Figura 12. Diagrama de flujo de la fase 3 de la técnica propuesta 105

Índice de Tablas

Tabla 1. Tasas de detección y protección de los antivirus 22

Tabla 2. Características intrínsecas propias del malware ransomware. 80

Tabla 3. Caracterización de las técnicas para la detección del malware ransomware 86

Tabla 4. Métricas para la evaluación de los clasificadores 102

Lista de Anexos

Anexo A. Fuentes sobre las características intrínsecas propias del malware ransomware..... 126

Abreviaturas

IA	Inteligencia Artificial
MW	Malware
BBDD	Base de Datos
Spyder	Scientific Python Development Environment
Sk-Learn	Scikit-Learn
SO	Sistema Operativo
IoT	Internet de las cosas
FPGA	Hardware basadas en matrices de puertas programables
ASIC	Circuitos integrados de propósito específico
ANN	Clasificación basada en redes neuronales
IDS	Sistema de detección de intrusos
API	Interfaz de programación de Aplicaciones
CSV	Valores separados por comas
JSON	Notación de objetos en JavaScript
KERAS	Biblioteca de Redes Neuronales de Código Abierto escrita en Python
	Lenguaje de Programación Interpretado Orientado a Objetos
TENSORFLOW	Biblioteca de código abierto para aprendizaje automático
DLL	Biblioteca de enlace dinámico
PE	Ejecutable portable
SVMs	Máquinas de soporte vectorial
RF	Random Forest
CPU	Unidad central de procesamiento.

1. Introducción

El malware, es un término que hace referencia a la expresión inglesa "*Malicious software*", la cual traduce software malicioso y es entendida como una amenaza persistente y avanzada que afecta las infraestructuras tecnológicas de forma crítica en diversas escalas, desde un equipo de uso personal hasta equipos tecnológicos en empresas públicas y privadas de los diferentes sectores productivos, educativos y gubernamentales. Un malware es creado por cibercriminales y su propósito es acceder a una computadora o servidor con el fin de afectar la máquina o robar la información, obteniendo así beneficios financieros (Estrada, 2018; Trigo, et al., 2017).

Dentro del amplio conjunto de malware creados por los cibercriminales, se encuentra la categoría denominada ransomware, un programa malicioso que "secuestra" información, es decir, encripta o bloquea los ficheros almacenados en un equipo informático con la finalidad de solicitar un rescate económico a la víctima para recuperar el acceso a sus archivos (Estrada, 2018). Por esta razón y por el continuo lanzamiento de nuevas formas de malware, que se alojan en la red a la espera de una oportunidad de infección, las empresas y las entidades continuamente deben mejorar sus técnicas de recuperación de datos y ciberseguridad.

A medida que avanza la tecnología y el procesamiento de los datos, se va dando la bienvenida a nuevas generaciones como: la 5G[1] y la Internet de las Cosas (IoT)[2], entre otros[3]; de

[1] La expresión 5G hace referencia a la quinta generación de redes móviles, las cuales permiten una navegación con mayor velocidad, disminuyendo el tiempo de respuesta de la red y aumentando el número de dispositivos conectados a una misma conexión (Flores, 2019).
[2] La Internet de las Cosas (IoT, por su sigla en inglés), se comprende como un sistema de "dispositivos de computación interrelacionados, máquinas mecánicas y digitales, objetos, animales o personas que tienen identificadores únicos y la capacidad de transferir datos a través de una red, sin requerir de interacciones humano a humano o humano a computadora" (Wigmore, 2021, párr. 1).
[3] Big Data, Inteligencia Artificial, fábricas inteligentes, etc.

ahí, que los procesos de seguridad informática están en continua mejora y sus estrategias y herramientas emplean los beneficios de estas nuevas tecnologías, como la inteligencia artificial (en adelante IA), para contrarrestar los riesgos potenciales y mejorar su efectividad.

Sin embargo, en el contexto de las comunicaciones 5G y la IoT, las tasas de transferencia alcanzadas son altas, hasta los 10 GBps (gigabytes por segundo) (Flores, 2019), y junto con un gran volumen de datos que se mueven por las redes de comunicación, hacen que la detección de malware-ransomware se convierta en un reto. Este se complejiza aún más debido a la creciente diversidad de este malware y que las reglas o técnicas de búsqueda de malware exigen su aplicación a cada paquete, dejando poco tiempo para examinar rigurosamente cada paquete que circula por la red. Un ejemplo de este fenómeno se evidencia en el estudio de Maimó (2019), quien expresa que:

> Cuando evaluamos el volumen de paquetes, que las actuales herramientas de inspección profunda pueden gestionar, nos encontramos con que la conocida Snort soporta redes cableadas de hasta 1 Gbps, empezando a descartar paquetes debido a sobrecarga a partir de 1,5 Gbps. Esto ha provocado la aparición de soluciones de hardware basadas en matrices de puertas programables (FPGA) o circuitos integrados de propósito específico (ASIC), que permiten trabajar con velocidades de hasta 7,2 Gbps. Aun así, estas velocidades quedan lejos de las que nos esperan en el futuro cercano. Debido en parte a esto, las soluciones de detección basadas en IDS (Intrusion Detection System) han tenido que evolucionar y pasar de analizar paquetes de red a analizar flujos de tráfico de red por medio de novedosas técnicas basadas en la inteligencia artificial. Por ejemplo, un modelo de red neuronal basada en bloques usada en un IDS basado en anomalías en flujos pudo trabajar con tráfico a 22 Gbps usando FPGAs. (p. 2).

Debido al alto flujo de información a través de las redes privadas e internet, y para asegurar la información que se encuentra almacenada en discos duros, bases de datos locales y en la nube; desde la ciberseguridad se requiere el uso de nuevos y mejores métodos para proteger esa información, tales como, las herramientas o técnicas que hacen uso de métodos de IA y la ingeniería inversa, para la detección de malware-ransomware, como lo hacen las herramientas de los antivirus, pero de forma más intuitivas y efectivas. En lo concerniente con la IA, Ruiz (2019), destaca el desarrollo y la consolidación de múltiples soluciones para la detección de malware mediante la implementación de técnicas basadas en Machine Learning (aprendizaje automático, en adelante ML), es decir, un conjunto de algoritmos "capaces de aprender de forma autónoma (…), capaces de extraer modelos o utilizar unos datos de entrenamiento para posteriormente poder predecir unos resultados" (p. 12).

Ahora bien, se ha identificado que los estudios corporativos basados en la eficacia que tienen las nuevas técnicas para la detección de ataques ransomware son pocos y/o escasos (Zufiaurre, 2019), por lo cual, con este proyecto se propone una nueva técnica para detectar malware ransomware usando la inteligencia artificial y así, contribuir a la seguridad informática, buscando, además, abrir puertas para continuar investigando sobre el funcionamiento de estas técnicas frente a los ataques cibernéticos. Entre otras acciones para la construcción de esta propuesta, se realizó un análisis de referentes bibliográficos enfocados en las técnicas de IA que aplican algoritmos de ML para la detección o caracterización del malware, haciendo un énfasis en el ransomware, los cuales dan cuenta de las bondades y los altos niveles de efectividad de las técnicas basadas en ML, para la detección y clasificación de este tipo de virus.

Este informe reporta los procedimientos y resultados investigativos obtenidos con el desarrollo del presente estudio, así como la técnica de detección propuesta; encontrando en el

siguiente capítulo el marco del proyecto, donde se explica el problema de investigación que origina la ejecución del mismo, junto con los antecedentes e hipótesis que permiten establecer el enfoque del proyecto. En el capítulo tres se formulan los objetivos, general y específicos que se cumplieron a lo largo de esta investigación, continuando con el marco metodológico (capitulo 4), donde se define la metodología empleada para la selección y análisis de la información requerida. Posteriormente, se halla el capítulo cinco en el que se describen y exponen los referentes teóricos y conceptuales implicados con el tema de este estudio, con énfasis en las técnicas de ML y el malware ransomware. Se continúa con el capítulo seis de desarrollo del proyecto, donde de forma explícita, se da cumplimiento a cada uno de los objetivos específicos planteados con esta investigación. Por otra parte, se tiene el capítulo siete con los resultados y la discusión, seguido de los apartados sobre el impacto esperado, las conclusiones y se finaliza con las recomendaciones futuras.

2. Marco del Proyecto

2.1. Definición del Problema

Las amenazas de malware están evolucionando de una forma precipitada a nivel mundial en muchos frentes, como son: los usuarios, dispositivos, aplicaciones y especialmente, se considera con el pasar de los años, que serán más vulnerables las redes sociales, el buzón de correo electrónico, redes empresariales y financieras (Romero et al., 2018).

Efectivamente, en años recientes se ha incrementado de forma acelerada, a nivel nacional e internacional, el número de personas y organizaciones que han sido víctimas de las amenazas con malware debido a la creciente cantidad de cibercriminales que han encontrado en los ciberdelitos una alternativa, que, si bien es de alto riesgo jurídico por la posibilidad de ser judicializados, también ha sido lucrativa para ellos. Así mismo, los efectos infecciosos y negativos de los malware se han incrementado por la multivariedad de malware que surge en el mundo diariamente y sus nuevas modalidades de ataque; los ciberdelicuentes comprenden que en la red viaja mucha información de vital importancia para los usuarios y las empresas, y que algunos de ellos están dispuestos a pagar por el rescate de esa información, si se perdiese o cayese en malas manos (Estrada, 2018).

Ante esta coyuntura global, algunas empresas de seguridad aseveran que particularmente el ransomware ha sido uno de los malware que ha provocado ataques masivos contra sus víctimas, dentro las cuales se encuentran las empresas públicas y privadas; lo que se explica por un incremento del 46% en el número de sus variantes y por las facilidades que tienen para el ataque objetivo contra las infraestructuras críticas, logrando bloquear los sistemas de control de las empresas.

En el ámbito internacional, Vivanco-Toala et al. (2020), han referenciado que entre los años 2013-2015, el número de ataques con ransomware a: los usuarios, a las grandes empresas y a las Pymes, se incrementó en un 270%, causado principalmente por vulnerabilidades en sus sistemas de ciberseguridad. Sumado a esto, "en el año 2016, se detectaron 372.602 ataques de ransomware (…) de los cuales el 17% iba dirigido a todo tipo de empresas grandes y pymes, donde el mayor porcentaje de ataques se evidencia para los consumidores o usuarios" (p. 73).

Por otra parte, se ha encontrado que durante los años 2019-2020 se registra una disminución en los ataques con ransomware contra los usuarios, pasando de 1'537.465 a 1'091.454 ataques, con un descenso equivalente al 29%, siendo WannCry el ransomware que más afecta a los usuarios; esta disminución posiblemente se ha originado porque las comunidades de ciberseguridad en el mundo han cobrado mayor relevancia en estos años. No obstante, también se halló que para el año 2020 hay un gran aumento de los ataques de ransomware dirigido[4], los que se realizan contra una víctima elegida con el objetivo de extorsionar, con un crecimiento del 767%, lo que deja a las empresas y grandes organizaciones en un considerable estado de vulnerabilidad, debido a que, siendo los objetivos principales de los ataques, los cibercriminales buscan que dichos ataques sean cada vez más sofisticados, profesionales y destructivos (Data Center Market, 2021).

La organización Kaspersky, a través de Aver (2020), registró en el año 2020 un listado de empresas que fueron víctimas de ataques con ransomware, encontrándose dentro de este, empresas tecnológicas de gran envergadura, como el proveedor de servicios informáticos y multinacional estadounidense Cognizant, la cual fue "víctima de un ataque del popular *ransomware* Maze. Los

[4] Los ataques con ransomware, suelen realizarse a objetivos de alto perfil, como empresas, organismos públicos estatales y municipales, y organizaciones sanitarias. Estos ataques son mucho más sofisticados (compromiso de red, reconocimiento y persistencia, o movimiento lateral) e implican un pago mucho mayor (Data Center Market, 2021).

clientes de la empresa utilizan su *software* y servicios para brindar soporte al trabajo remoto de los empleados, cuyas actividades se vieron interrumpidas" (párr. 8). Este suceso es un claro indicador que incluso empresas con gran infraestructura tecnológica y altos niveles de seguridad no están exentas a un ataque de ransomware.

En el contexto nacional, Ceballos et al. (2019), presentan un estudio sectorial mediante el cual registraran "las cifras y modalidades de los ciberdelitos en 2019 y las tendencias que [enfrentarían] las empresas colombianas y los ciudadanos en 2020" (p. 4); la información fue obtenida a partir de las denuncias y reportes efectuados por las empresas y los ciudadanos al Centro Cibernético Policial (CECIP). En este informe, se reportó que: el hurto por medios informáticos es el delito informático más denunciado en Colombia, seguido de la violación de los datos personales (robo de identidad), en tercer lugar, está el acceso abusivo a un sistema informático, en cuarto lugar, se encuentra la transferencia no consentida de activos y en quinto lugar, está el uso de software malicioso (ransomware).

Con respecto a este último, los investigadores identificaron que es un delito cuya comisión se ha incrementado en los años recientes en el país, siendo las Pymes, las compañías más afectadas por los ransomware, debido a sus bajos niveles de ciberseguridad. Principalmente, se han detectado cinco tipos de ransomware: de cifrado Lock Screen Ransomware o WinLocker, Master Boot Record (MBR), Ransomware alware, de cifrado de servidores web y de dispositivos móviles; los cuales suelen ser activados por los usuarios colombianos, a través de los enlaces que llegan con correos electrónicos y su respectivo redireccionamiento hacia sitios web maliciosos, desde donde descargan el malware que afecta sus documentos. Se reportó que, una de las principales tendencias para el año 2020 en el país es el uso de inteligencia artificial y malware, por parte de los ciberdelincuentes, para facilitar la identificación de víctimas potenciales y los sistemas de

detección usados por los usuarios. Entre otras recomendaciones de la investigación, se considera el desarrollo de mecanismos perimetrales de protección (Ceballos et al., 2019).

De acuerdo con el panorama actual con el malware ransomware, las organizaciones de ciberseguridad y las empresas en general, hacen un llamado enérgico para que los usuarios y las empresas, especialmente las Pymes: adelanten y ejecuten las medidas necesarias para incrementar la ciberseguridad de sus sistemas locales y en la nube; refuercen el cifrado de los documentos con nuevas técnicas; que se informen y estén alerta de las crecientes y más sofisticadas acciones de ingeniería social que usan los cibercriminales para afectar a sus víctimas; y que aprovechen las ventajas y capacidades que ofrecen la automatización y los sistemas de inteligencia artificial ya que pueden ser entrenados para detectar comportamientos anómalos y también para enfrentar las nuevas modalidades que surgen con respecto a los ataques con ransomware (Organización Deloitte, 2021). Todas las precauciones son necesarias, teniendo en cuenta que en numerosos casos "el impago del rescate no da como resultado la destrucción de la información, sino su publicación en fuentes abiertas o su venta en subastas (cerradas)" (Aver, 2020, párr. 3), afectando directamente el derecho a la privacidad y el secreto empresarial.

Como respuesta a los continuos y destructivos ataques de malware que se presentan en la actualidad, los antivirus[5] y firewalls[6] actuales combinan una base de datos de firmas maliciosas con una gran variedad de enfoques que incorporan técnicas avanzadas a la detección de virus maliciosos, en especial, el malware ransomware (Maimó, 2019), como los sistemas desarrollados

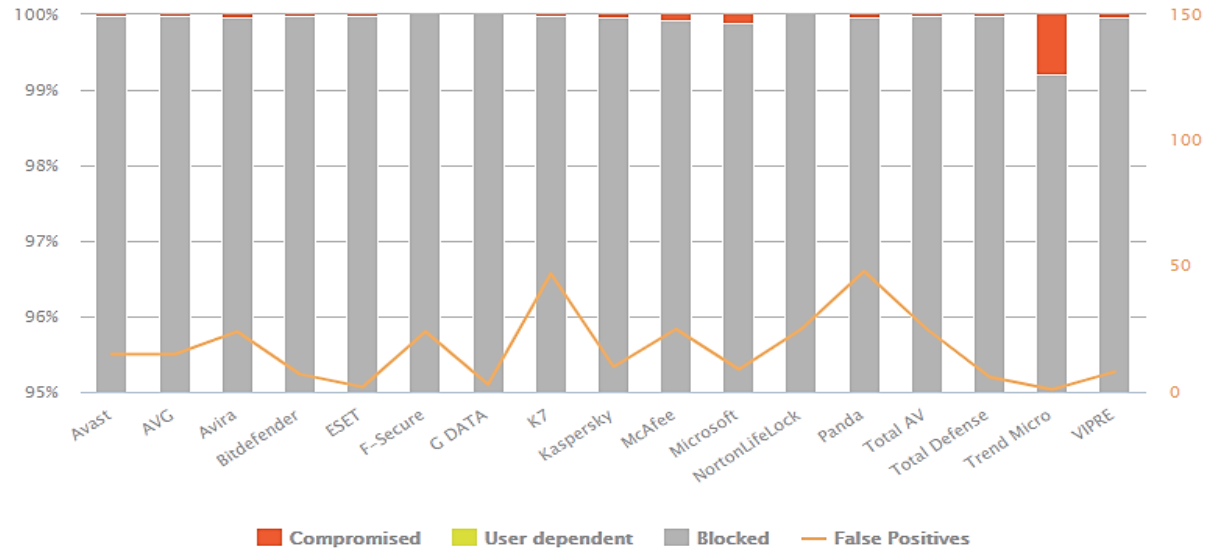
[5] De acuerdo con Padilla (2010), un antivirus "es un programa de computadora, que mediante un escaneo de archivos tiene como objetivo la detección, identificación y eliminación de *malware"* (p. 1).
[6] Un firewall (cortafuegos en español) "es un dispositivo de seguridad de red que monitorea el tráfico de red entrante y saliente y decide si permitir o bloquear tráfico específico según un conjunto definido de reglas de seguridad" (Cisco Systems, Inc., s.f., párr. 1).

a partir de técnicas de programación heurística[7] con las cuales se busca anticipar el descubrimiento del malware, ya que es capaz de bloquearlo sin tener en su base de datos conocimientos absolutos de que ese programa sea un software malicioso, con la desventaja de una alta tasa de falsos positivos. Del mismo modo, la técnica de análisis de comportamiento analiza todos los movimientos que ejecuta una aplicación, determinando si se trata de un comportamiento no deseado, basándose en el consumo del 100% en la CPU, consumo al máximo de la memoria RAM y muchas operaciones de lectura/escritura en HDD de la máquina que, al igual, podrían ser falsas alarmas (Herrero, 2018).

A continuación, en la figura 1 se visualiza la eficacia de cada uno de los antivirus que se utilizan en la actualidad y su reporte con respecto a la detención del malware.

Figura 1.

Eficacia de un antivirus, según la detección de malware y falsos positivos



Fuente: tomada de Sánchez (2020).

[7] Algunas técnicas para el análisis heurístico con antivirus, son: emulación de archivo, análisis del archivo y Detección de Firma Genérica (Firma Digital), etc.

Los datos de la figura 1 revelan que si bien cada uno de los antivirus referenciados registra tasas de falsos positivos relevantes, principalmente con respecto a los antivirus K7 y Panda, también es posible indicar que estos resultados no representan un peligro para los usuarios, sino que más bien son una incomodidad para los mismos, porque implican falsas alertas. Sin embargo, se evidencia además, que las tasas de falsos negativos, es decir, el número de muestras que comprometieron el sistema con una detección no producida, para un archivo que sí es malicioso, denominado en esta grafica como "compromised"; son considerablemente altas, si se tiene en cuenta que una sola muestra puede tener un efecto fatal para la maquina; lo que se presenta en mayor medida con el antivirus Trend Micro, Microsoft y McAfee, siendo un valor nulo en este caso, con respecto a F-Secure y G DATA.

Tabla 1.

Tasas de detección y protección de los antivirus

	OFFLINE Detection Rate	ONLINE Detection Rate	ONLINE Protection Rate	False Alarms
Avast	94.2%	99.6%	99.98%	15
AVG	94.2%	99.6%	99.98%	15
Avira	90.4%	96.4%	99.97%	24
Bitdefender	96.1%	96.1%	99.98%	7
ESET	94.0%	94.0%	99.99%	2
F-Secure	90.4%	96.4%	100%	24
G DATA	96.2%	96.2%	100%	3
K7	93.5%	93.5%	99.99%	47
Kaspersky	81.9%	92.3%	99.97%	10
McAfee	67.0%	99.7%	99.93%	25
Microsoft	70.5%	85.9%	99.88%	9
NortonLifeLock	85.3%	99.3%	100%	25
Panda	56.9%	91.8%	99.96%	48
Total AV	90.4%	96.4%	99.99%	25
Total Defense	96.1%	96.1%	99.99%	6
Trend Micro	52.3%	94.4%	99.20%	1
VIPRE	96.1%	96.1%	99.97%	8

Nota: tomada de Sánchez (2020).

Prosiguiendo con lo anterior, se observa que en términos generales, la eficacia de los antivirus evaluados en la tabla 1 en materia de bloqueos es muy alta, especialmente, en cuanto a los antivirus F-Secure, G DATA y NortonLifeLock, cada uno con una tasa de protección del 100%, aunque también se evidencia que el porcentaje de bloqueos de los antivirus fuera de línea decae considerablemente, porque necesitan tener conexión a su motor de firmas en línea; siendo mayor esta variación en lo concerniente con las disminuciones que revelaron los antivirus Trend Micro (-42,1%), Panda (-34,9%), McAfee (32,7%) y Microsoft con una reducción del 15,4%. Con respecto a la tasas de detección en línea, los resultados más altos se registran con los antivirus McAfee, Avast, AVG y NortonLifeLock. Así las cosas, una dependencia tan grande a sus bases de datos de firmas es una debilidad de los antivirus actuales que se podría mejorar considerablemente incluyendo técnicas de IA.

Estos problemas hacen necesaria la búsqueda de una técnica mucho más efectiva que combine las ventajas de los dos métodos anteriormente descritos con técnicas basadas en algoritmos de la IA, tales como los algoritmos de aprendizaje automático y aprendizaje profundo; los cuales según Lazzeri (2021) estudian patrones complejos a partir de la transformación de los datos de entrada, en varias capas de salida y ocultas, mejorando las tareas de detección y clasificación con la experiencia. Siendo de las técnicas más novedosas, las técnicas de análisis basados en IA se anticipan a malware potenciado por IA, además de la detección de malware tradicional, aumentando la tasa de detección de códigos maliciosos desconocidos al realizar simultáneamente análisis estático y análisis dinámico, basados en motores de aprendizaje automático y profundo además de motores de vigilancia de IA en tiempo real (Ceballos et al., 2019). Esta combinación ha revelado buenos resultados de detección dinámica, aunque con la

desventaja de que son técnicas que necesitan de un entrenamiento, a partir de bases de datos y repositorios de malware-ransomware con atributos determinados.

En vista de lo anterior, se evidencia que si bien en la actualidad, existen diversos antivirus y firewall para la detección de malware ransomware, también se identifica que los altos niveles de falsos negativos generados en su detección offline, distan de ser realmente efectivos con este propósito y dejan a los usuarios y empresas vulnerables ante posibles ataques con software malicioso. Entre otros asuntos, esta problemática se origina con los requerimientos de tiempo, es decir, capacidad inmediata de reacción, y con los requerimientos de detección en las diversas formas de presentación del ransomware.

2.2. Justificación del Problema

Desde la literatura, se identificó la deficiencia e ineficacia de los antivirus frente al malware-ransomware, demostrando una gran brecha con respecto a los falsos negativos en materia de detección y detención (Herrero, 2018). No obstante, con el desarrollo y la vertiginosa evolución en los avances tecnológicos, se han creado nuevos y mejores sistemas, herramientas y métodos de seguridad (como aquellos basados en ML), para la prevención, detección temprana, clasificación y protección de la información digital; los cuales, junto con buenas prácticas de concientización de los usuarios en temas de seguridad, permiten la optimización de sus respuestas frente a posibles ataques con malware ransomware y además contribuyen a reducir las consecuencias negativas generadas por un ataque de malware (ESET Latinoamérica, 2015).

Al respecto, diversos autores han comprobado y reconocen que el uso e implementación de técnicas de ML es una alternativa viable para la identificación y detección automática de diversas formas de malware, entre estos el ransomware, porque: revelan un menor tiempo para la

ejecución de ambos procesos y una mejora en el umbral de detección fuera de línea (Becerra & Vargas, 2019); también, facilitan la labor de los analistas forenses mediante la automatización de los procesos de obtención de las muestras de ransomware (Benavides & Roa, 2018); favorecen la ejecución de acciones de monitoreo y la toma de decisiones anticipadas para evitar comprometer funciones y actividades críticas de las instituciones financieras, con respecto a posibles infecciones por malware (Mayorga, 2017); y contribuyen a que los sistemas sean más dinámicos en el análisis y procesamiento de datos, gracias al uso de algoritmos del ML (Márquez, 2017); entre otros beneficios.

Si bien desde la literatura, se identifican aportes relevantes con respecto a la importancia que ha suscitado la utilización de herramientas de ML para la prevención y detección de malware, también se ha encontrado una deficiencia en la cantidad de publicaciones enfocadas específicamente en la aplicación de estas técnicas para el caso del malware ransomware; de ahí que autores como Becerra y Vargas (2019) formulan la creciente necesidad de continuar con el desarrollo de investigaciones acerca de la detección de diversos incidentes de seguridad que puedan beneficiarse de algoritmos de ML.

De acuerdo con lo anterior, la importancia de este estudio en profundidad sobre el malware-ransomware radica en la posibilidad de clarificar de forma extensiva su origen y su propagación en diversos escenarios asociados a los usuarios que se ven afectados. La comprensión de estos aspectos del ransomware permitirá, además, un acercamiento a la seguridad de las máquinas, de las redes y a la protección de los datos, junto con su relevancia en la protección de la privacidad del usuario. De igual forma, esta investigación es importante, porque mediante la compresión de los conceptos asociados con el malware ransomware y los procedimientos de detección

adelantados con el uso de técnicas de ML, se propone una técnica que redunde en un proceso de detección de ransomware efectivo.

Adicionalmente, proponer una técnica de detección basada en ML contribuirá a la obtención de mejores resultados en cuanto a la detección del malware ransomware, mejorando aspectos como: la tasa de falsos negativos, la anticipación a malware potenciado por IA, el ajuste dinámico del sistema para la detección de códigos maliciosos desconocidos, potencializando el análisis estático con un análisis dinámico basados en aprendizaje automático, además de motores de vigilancia que observen los canales de navegación, comunicación y archivos de fuentes poco confiables para evitar los problemas asociados un secuestro de información por ransomware.

2.3. Formulación del Problema

Del problema descrito y la revisión de antecedentes se formula la siguiente pregunta de investigación:

¿Cuáles son las características que debe tener una técnica de detección de malware-ransomware usando la inteligencia artificial?

3. Marco Contextual

3.1. Aspectos Generales.

Según el Centro Cibernético de la Policía Nacional de Colombia (Ceballos et al., 2019) para el periodo 2019 -2020, la mayor violación fue para el artículo 269i, el cual consiste en hurto por medios informáticos y semejantes. El segundo delito informático con mayor incidencia en Colombia, con 8.037 casos, es la violación de datos personales o robo de identidad; el tercer delito más denunciados es acceso abusivo a sistema informático, con 7.994 casos; el cuarto lugar con 3.425 casos es la transferencia no consentida de activos; y en quinto lugar se encuentran los ataques por malware – ransomware con 2.387 casos denunciados.

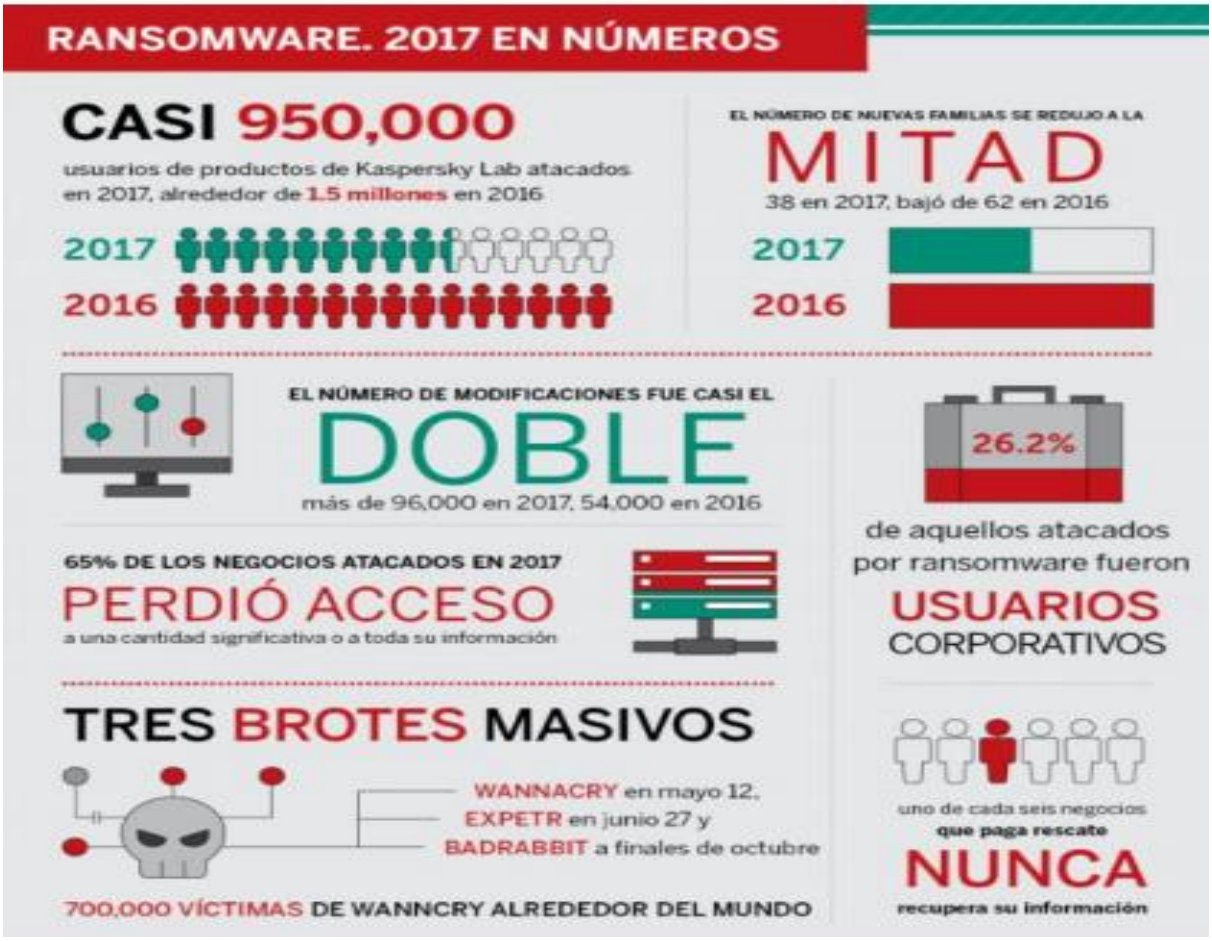
Colombia recibió el 30% de los ataques de ransomware en Latinoamérica en el año 2020, seguido por Perú (16%), México (14%), Brasil (11%) y Argentina (9%), según lo reporta el *Informe de las Tendencias del Cibercrimen en Colombia 2019-2020*. Las PYMES fueron el blanco preferido por los atacantes, puesto que sus niveles de seguridad suelen ser más bajos; no obstante debido a los vectores de infección que utilizan los cibercriminales como entrada del ransomware, tales como: correo electrónico fraudulento, noticias de alerta falsas en ventanas emergentes, documentos con macros, phishing[8], Cross-Site Scripting – XSS[9] , entre otros métodos que apelan a la poca precaución en la navegación de los usuarios, pueden llegar a afectar a personas y empresas de diferentes sectores sin discriminación.

[8] Los ataques de phishing son correos electrónicos, llamadas o mensajes de texto maliciosos que engañan a los usuarios para que cedan sus credenciales de cuenta. El remitente se hace pasar por una entidad acreditada. Este atrae a los usuarios para que proporcionen información confidencial, incluidos los detalles de la tarjeta de crédito y las contraseñas.

[9] XSS es un ataque de inyección en el que un hacker inyecta un script o código malicioso en el contenido de un sitio web de modo que una vez se accede este queda instalado y puede ejecutarse desde el navegador de los usuarios.

Figura 2.

Estadísticas ransomware en usuarios corporativos



Fuente: tomada de Kaspersky Lab (2018).

De acuerdo con los datos de la figura 2, es posible dilucidar los ataques de ransomware son un exponente de vulneración a los derechos cibernéticos y de la privacidad; el ataque de los ransomware como: WannaCry, CryptoLocker, Petya, Expert y Badrabbit, entre otros, afectan a usuarios corporativos de todo el mundo. Las entidades Industriales son las que mayormente se ven afectadas en Colombia, ya que, para esa misma anualidad, por lo menos 446 empresas denunciaron

ser víctimas de ciberataques. La seguridad busca reducir los riesgos hasta un nivel aceptable, ya que es difícil asegurar que se apunte a una solución ideal para evitar todos los peligros, es decir, en todos los ámbitos la seguridad busca reducir los riesgos y buscar actividades encaminadas a proteger de algún tipo de peligro (Kaspersky Lab, 2018). Nuevas variantes de ransomware se cobran víctimas cada día y por ello sigue siendo una preocupación a nivel global.

3.2. Antecedentes

En esta sección, desde la revisión de artículos resultados de investigación y trabajos de grado, consultados en repositorios institucionales y bases de datos indexadas como Science Direct y la IEEE, se da cuenta de la perspectiva y conclusiones que los autores tuvieron en sus investigaciones, brindando una mejor claridad teórica para abordar esta propuesta de investigación, cuyo objetivo de estudio es identificar las técnicas de ML utilizadas en la detección de malware ransomware y los resultados obtenidos con su implementación.

En este orden de ideas, el trabajo realizado por García (2020), para obtener el título en Ingeniería de Software, llamado *Auditoría automatizada basada en un sistema de detección de vulnerabilidades y en la explotación controlada de amenazas software* de la Universidad de Málaga – España; el cual tiene como objetivo realizar una aplicación con la que pueda dotar al usuario con una herramienta con la que pueda defender su información. La aplicación indica los procesos que se estén ejecutando actualmente en el sistema. Del mismo modo, se busca ejecutar un análisis de la red en cuanto a tamaño, puerto y procedencia del paquete que se esté iniciando mediante ML. Así, el usuario puede darse cuenta qué dispositivos están conectados a su red, a través de *Universal Plug and Play* (UPnP) al punto de acceso. La metodología que utilizó el autor fue SCRUM, "con el objetivo de realizar un desarrollo iterativo para identificar posibles errores

de concepto durante la realización del proyecto, de forma que los cambios no introduzcan una complejidad alta en el mismo" (p. 19), aunque presenta una desventaja con respecto al usuario inexperto.

Por otra parte, el autor Maimó (2019), en su tesis doctoral: *Detección de bonets y ransomware en redes de datos mediante técnicas de aprendizaje automático*, presentada en la Universidad de Murcia en España, identificó tres puntos débiles que presentan los problemas de detección: a) el equilibrio necesario entre la efectividad de la detección de amenazas y la velocidad a la que se pueden examinar los datos recogidos en las redes de datos modernas; b) la necesidad de poder detectar dichas amenazas incluso cuando los datos viajan cifrados; y c) la dificultad de detectar nuevas versiones de malware aunque sean variaciones de una familia ya conocida. El objetivo de este autor fue "investigar la forma de aplicar métodos de aprendizaje automático a la detección de anomalías en redes de datos con restricciones" (p. 5); para lo cual determinó su metodología en el estudio del estado del arte de los sistemas de detección basados en el aprendizaje automático de las redes de bases de datos. Su enfoque estuvo basado en flujos y se estudiaron los requerimientos computacionales de varios métodos de ML, con el fin de encontrar el más adecuado. La desventaja de este estudio, se origina en la imparable expansión del internet y la mayor adopción del cifrado en el tráfico de red.

En continuidad con lo anterior, los investigadores Lu et al. (2019), para el desarrollo del trabajo *New era of deeplearning-based malware intrusion detection: the malware detection and prediction based on deep learning*, de la University of Chinese Academy of Sciences, (Universidad de la academia de ciencias de China), realizaron una investigación con el objetivo de observar cómo se comporta el malware para luego clasificarlos, a través de algoritmos tradicionales de la máquina y los algoritmos de red neuronal profunda, para luego comparar el resultado con el

sistema de clasificación de malware basado en reglas, en dos tipos diferentes de conjunto de tareas. Después de hacer la comparación, se encontró que el aprendizaje profundo se desempeña mejor en cuanto a precisión y versatilidad. Es más seguro analizar los datos mediante un campo de entrenamiento que hacer un campo de búsqueda profunda. También se construyó un modelo de GAN basado en un texto y en una imagen de datos con software malicioso y diseñar una especie de nueva arquitectura de predicción de malware que muestre cierta viabilidad en las bases de datos probadas.

Otro aporte identificado desde la literatura, es el trabajo de grado efectuado por García (2020), titulado *Auditoría automatizada basada en un sistema de detección de vulnerabilidades y en la explotación controlada de amenazas software;* mediante el cual, se reconoce la relevancia de las herramientas Big Data y analíticas en el tratamiento y conocimiento de los datos, de ahí que la finalidad con esta investigación es predecir los computadores que resultarán infectados por malware en una institución financiera colombiana, mediante el diseño y la implementación de la Prueba de Concepto (POC), para lo cual se contó con el apoyo operativo del fabricante Hewlett-Packard y el proveedor itPerform, quienes de forma conjunta utilizaron dos servidores: 1) Servidor Windows 2008 R2, con las analíticas ETL y ARIMA, incluidos en el software alterix (preparación de datos) y tableau (presentación de resultados) y 2) Servidor Linux CentOs 7.0 con el programa Big Data HP VERTICA. Los datos analizados corresponden a los datos log de las acciones realizadas por el antivirus instalado en cerca de 4.000 computadores de la Institución Financiera, los cuales se encuentran alojados en la herramienta de seguridad SIEM (Security Information and Event Management).

Se contempla el uso del modelo predictivo de “Decision Tree”, pero este solo revela cual es el computador con más cantidades de eventos que posiblemente informará el antivirus y también

el modelo de "Naive Bayes", aunque este solo predijo los computadores agrupados por un tipo de malware; finalmente, se utiliza el modelo predictivo ARIMA con el que se logra la predicción con dos semanas de anterioridad de los computadores que presentarían una infección con malware y con mayor rapidez en términos de tiempo, que los modelos tradicionales utilizados en esta institución; datos que se verificaron comparando la predicción y la información real semanas después, encontrando un alto grado de correspondencia entre ambos informes. Además, se alcanza un comportamiento de descenso y de tendencia cero para los computadores que fueron infectados por malware. Se comprueba que el desarrollo de un modelo predictivo para detectar de forma anticipada los computadores que serán infectados por malware, favorece la ejecución de acciones de monitoreo y la toma de decisiones para evitar comprometer funciones y actividades críticas de las instituciones financieras (García, 2020).

Por otro lado, el autor Carmona (2016), en un tutorial presentado por el Dpto de inteligencia artificial, ETS de ingeniería informática de la universidad nacional de Madrid – España, con el propósito de dar a conocer los nuevos cambios o alcances que han obtenido las máquinas de vector soporte (SVM) con la versión (16/11/2016) iniciada desde (17/112013), llamado *Tutorial sobre Máquinas de Vector Soporte*, plantea como objetivo principal *la realización de una introducción al mundo de SVM*, aplicadas a resolver tareas tanto de clasificación como de regresión, teniendo presente que a la primer tarea se restringe al caso de clasificación binaria, atendiendo al tipo de separabilidad de los ejemplo de entrada considerados en distintas opciones.

Estas opciones son discriminadas en los siguientes casos y en el mismo orden, para el primer caso sería el ejemplo de entradas separables, para el segundo caso, aunque es con ruido, sería el ejemplo de entradas cuasi-separables linealmente y seguidamente, se considera el caso de ejemplo no-separable linealmente y finalmente, las SVMs aplicadas a la tarea de regresión lineal

como no lineal. Este proyecto proporciona las bases de aprendizaje al mundo de las máquinas de vector soporte, con el fin de poder hacer clasificaciones binarias o poder resolver tipos de problemas de regresión, agrupamiento y multiclasificación aplicable a las redes neuronales y a la propuesta planteada en este proyecto. El tutorial al final hace recomendación de algunas herramientas software de uso libre para un mejor aprendizaje y en las que podríamos empezar a experimentar (Carmona, 2016).

3.3. Hipótesis

La vulneración de la privacidad aumenta de forma casi paralela a los descubrimientos de nuevas tecnologías, es por esto que los avances más actuales deben estar a la vanguardia en amenazas que tienen como único objetivo, el daño a cualquier entidad. A través del estudio del malware ransomware, su composición, la explicación de sus funciones en el ataque, su magnitud, sus principales objetivos y cómo intervenirlos a través de herramientas de la Inteligencia Artificial como el aprendizaje automático, se pueda proponer una técnica para su detección.

3.4. Objetivos

3.4.1. Objetivo General

Proponer una técnica de detección de malware-ransomware utilizando algoritmos de Machine-Learning.

3.4.2. Objetivos Específicos

- Identificar las técnicas que involucran algoritmos de Machine-Learning utilizadas para la detección de malware–ransomware.

- Analizar las características principales de las técnicas identificadas para la detección de malware–ransomware.

- Construir la propuesta de la técnica apoyada en algoritmos del Machine Learning para la detección de malware-ransomware.

4. Marco Metodológico

4.1. Metodología

Esta investigación tiene como objetivo principal proponer una técnica de detección de malware ransomware basada en el análisis comparativo de las técnicas actuales que utilizan algoritmos del Machine Learning como parte del proceso de caracterización, detección y detención del ransomware y que de alguna forma buscan contrarrestar los daños ocasionados por el ataque del software malicioso.

En correspondencia con lo anterior y según los referentes teóricos de Ricoy (2006) junto con Hernández, Fernández y Baptista (2014), la presente investigación se sitúa en el paradigma positivista, además, es un estudio con enfoque cuantitativo, no experimental; porque se logra la construcción del objeto de conocimiento en relación con el comportamiento del malware ransomware, conforme con las experiencias, los hechos observables y la información objetiva recopilada desde la literatura, en relación con los experimentos efectuados por otros autores, para la implementación de técnicas de Machine Learning en la detección de este tipo de software malicioso y los resultados recopilados al respecto. Todo este procedimiento sin la intervención directa del investigador, ni la manipulación deliberada de las variables de análisis, aunque basado fundamentalmente en la revisión, observación y análisis de esta temática, tal como ha sido abordada por otros investigadores.

Sumado a esto, se realiza un análisis documental, mediante el cual se identificaron las técnicas actualmente utilizadas, sus etapas, la contribución de los algoritmos de Machine Learning en estas, y las características de la tipología malware-ransomware que permiten su detección. Este proceso metodológico se efectuó, a través de la revisión y análisis de 35 publicaciones extraídas

de diversas fuentes bibliográficas, tales como: artículos de revistas indexadas resultados de investigación, trabajos de grado de repositorios institucionales nacionales e internacionales, así como informes y boletines técnicos sobre los avances y novedades en materia de malware ransomware. Es preciso clarificar que para la selección de los documentos estudiados, se tuvieron en cuenta los siguientes criterios de inclusión: (a) una temporalidad de publicación máxima de cinco años, comprendida entre los años 2016-2020 y (b) que por lo menos cumpliese con dos de las tres etiquetas definidas con esta finalidad, a saber: malware, ransomware y Machine Learning.

Tomando como base los métodos inductivos[10]- deductivo[11], a partir de la revisión de la literatura, el estudio de información técnica acerca del malware ransomware y de las técnicas de detección empleadas haciendo énfasis en las de aprendizaje automático, se elaboraron conclusiones especificas en materia de técnicas de Machine Learning para la detección y clasificación de los malware ransomware que permitieron plantear la técnica propuesta como resultado de la investigación.

4.2. Tipo de Investigación

Al retomar los aportes de Hernández et al. (2014), es posible señalar que se efectuó una investigación exploratoria, ya que se pretendió encontrar aspectos fundamentales de la detección de malware-ransomware que pudieran ser aprovechados por algoritmos de aprendizaje automático para su detección; se toma como referencia y punto de partida los estudios consultados durante la

[10] El método inductivo es una estrategia de razonamiento que se basa en la inducción, para ello, procede a partir de premisas particulares para generar conclusiones generales. En este sentido, el método inductivo opera realizando generalizaciones amplias apoyándose en observaciones específicas.
[11] El método deductivo es una estrategia de razonamiento empleada para deducir conclusiones lógicas a partir de una serie de premisas o principios. En este sentido, es un proceso de pensamiento que va de lo general (leyes o principios) a lo particular (fenómenos o hechos concretos).

revisión de literatura, antes y durante la construcción de este proyecto, apoyándose en el estado del arte, donde se enfoca en los siguientes tipos de estudio:

- **Identificar técnicas de Machine Learning existentes sobre detección de malware-ransomware:** Aplicando el método inductivo – deductivo en la identificación de las técnicas existentes actuales en la detección sobre el malware-ransomware, se llegaron a una serie de conclusiones con las que se tomaron decisiones, en cuanto a una propuesta de mejora en tiempo y efectividad de detección.
- **Analizar las características de las técnicas sobre la detección de malware-ransomware:** En continuidad, se aplicó el método analítico – sintético para analizar las características intrínsecas de los ransomware y de las técnicas que existen sobre la detención, tomando como base otras investigaciones elaboradas al respecto.
- **Proponer una técnica de detección de malware-ransomware usando las mejores técnicas de detección de acuerdo al análisis hecho:** En consecuencia, aquí se utilizaron los métodos inductivo, deductivo, analítico y sintético para elaborar la propuesta de la técnica de detección del malware-ransomware, basado en el estudio y análisis hecho de las herramientas de última generación como lo es IA, que es una simulación de procesos que realiza la inteligencia humana en máquinas convirtiéndola en una "máquina inteligente", es decir, aquella que imita las funciones cognitivas como percibir, razonar, aprender y resolver problemas (Zufiaurre, 2019).

4.3. Definición del Alcance

La presente investigación articula la propuesta de una técnica basada en algoritmos de ML que proporcionará una nueva forma en la detección y prevención del malware ransomware. No se incluyó la validación de la técnica propuesta con el desarrollo de este proyecto, por falta de recursos computacionales, se plantea hacer la validación de la técnica en un ambiente controlado como trabajo futuro.

5. Marco Teórico – Conceptual

En este capítulo se tratan las bases teóricas que se usaron en el desarrollo de este trabajo, partiendo de las premisas de seguridad informática. Se pretende conocer qué es y en qué consiste el malware ransomware; también se presenta un análisis profundo de todos sus comportamientos actuales y como las técnicas más utilizadas de Machine-Learning han contrarrestado sus acciones maliciosas, para así poder proponer una nueva técnica óptima con cortos tiempos de respuesta.

5.1. La Ciberseguridad

Llamada también seguridad de tecnología de la información, es el área relacionada con la informática y la telemática que se enfoca en la protección de la infraestructura computacional y todo lo relacionado con esta y especialmente en la información contenida en una computadora o circulante a través de las redes de computadoras. Para ello existe una serie de estándares, protocolos, métodos, reglas, herramientas y leyes concebidas para minimizar los posibles riesgos a la infraestructura o a la información. La ciberseguridad comprende software (bases de datos, metadatos, archivos), hardware, redes de computadoras y todo lo que la organización valore y signifique un riesgo (Paez, 2020).

Por lo tanto, los datos son valores, números, medidas, textos, documentos en bruto, la información es el valor de esos datos, es lo que aporta conocimiento. Los manuales de procedimientos, los datos de los empleados, de los proveedores y clientes de la empresa, la base de datos de facturación son datos estructurados de tal forma que se convierten en información, que aportan valor como compañía por tal motivo deben ser protegidos (Romero, y otros, 2018).

5.2. Malware ransomware

Ransom quiere decir “rescate” en inglés, y de hecho lo que hace es secuestrar los datos de un ordenador y pedir un rescate económico a cambio de liberarlo. Normalmente lo que hace es cifrar tus datos, y lo que te ofrecen a cambio del rescate económico es la clave para poder descifrarlos. Este tipo de programas puede acceder a tu ordenador por medio de un gusano informático u otro tipo de malware, y una vez cifre tus datos bloqueará tu ordenador mostrándote una pantalla de advertencia en la que se le informa al usuario que ha sido víctima del ataque. En esa pantalla se le muestra también la cantidad a pagar y el método de pago, que puede ser por SMS, Paypal o mediante bitcoins. Se trata de una de las amenazas que más está creciendo en los últimos años, éstos llegan por diferentes medios, en especial por correos electrónicos o mensajes sospechosos (Fernández, 2020). A continuación, en la figura 3, se muestra el esquema de funcionamiento de éste ransomware.

Figura 3.

Esquema de funcionamiento ransomware

Nota. Uso de analíticas para predecir los computadores afectados. Fuente: Mayorga (2017).

5.3. Delincuentes Informáticos

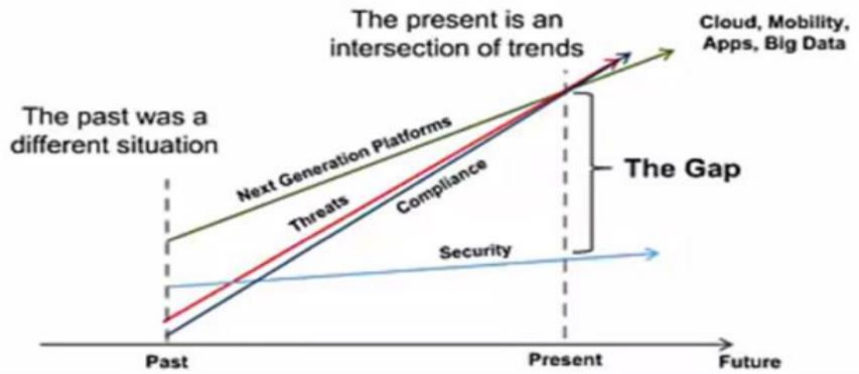
Los delincuentes informáticos ya no son sólo pequeños grupos de hackers malintencionados, sino que existen complejas organizaciones dedicadas a crear herramientas, malware y estrategias para llevar a cabo ataques cibernéticos. Incluso hay un verdadero mercado de software malicioso a medida, con el que los delincuentes pueden comprar y desarrollar herramientas altamente sofisticadas. Además, ya están usando la inteligencia artificial para mejorar la planificación y ejecución de sus ataques, por lo que las organizaciones están luchando constantemente para mejorar la protección de sus sistemas (Rey, 2019).

5.4. Brecha de Seguridad Informática

Al revisar extensa documentación en la web se ratifican e identifican las principales causas del problema antes mencionado y una excelente forma de explicar el por qué las herramientas de seguridad informática no han sido efectivas al 100% para controlar los problemas por malware (Mayorga, 2017).

Figura 4.

Brecha de seguridad informática



Nota. Uso de analíticas para predecir los computadores afectados. Fuente: Mayorga (2017).

Se utiliza el término “The Gap” o “Brecha de seguridad informática” para indicar la existencia de un problema ocasionado por un espacio existente entre los desarrollos de seguridad informática versus los avances tecnológicos para desarrollar amenazas informáticas, el cumplimiento de leyes para la protección de datos y las plataformas de siguiente generación que avanzan a grandes pasos, de forma incremental y se representan en la figura 4, con una pendiente mayor a 45º; mientras que la línea que indica los avances de la seguridad informática tiene una pendiente levemente inclinada, representando avances más lentos, debido a los tiempos de investigación de fabricantes de nuevas tecnologías, la comercialización de los productos, la aceptación por las empresas y su implementación, con el respectivo afinamiento y mantenimiento de la solución informática (Mayorga, 2017).

5.5. Tipos de Malware

Figura 5.

Tipos de Malware

Fuente: tomada de Mundaca (2020).

Como se evidencia en la figura 5, se han identificado seis familias de malware, los cuales han sido agrupados, según sus patrones de acceso y que se explican a continuación.

- **Un virus informático:** es un tipo de malware cuyo objetivo es alterar el correcto funcionamiento de un dispositivo. Lo hace infectando los ficheros de un ordenador mediante un código maligno, y su principal característica es que necesita de la intervención del usuario para ser ejecutado. Momento en el que toma el control con el objetivo de infectar un ordenador y propagarse (Mayorga, 2017).
- **El gusano informático:** es otro de los tipos de malware más comunes en la red, y su principal diferencia con los virus informáticos es que no necesita la intervención del usuario ni la modificación de ningún archivo existente para infectar un equipo. Por lo demás, tiene la característica de replicarse a sí mismo para expandirse por las redes a las que está conectado un dispositivo (Mayorga, 2017).
- **El spyware:** se trata de otro tipo de programa que se instala en tu equipo por sí sólo o mediante la interacción de una segunda aplicación que lo lanza sin que te des cuenta. Suelen trabajar a escondidas tratando de ocultar su rastro para que levantes la guardia y actúes con normalidad (Mayorga, 2017).
- **El troyano** tiene algunas semejanzas con los virus informáticos, pero su funcionamiento no es exactamente el mismo. Mientras que un virus suele ser destructivo, un troyano trata de pasar desadvertido mientras accede a tu dispositivo con la intención de ejecutar acciones ocultas con las que abrir una puerta trasera para que otros programas maliciosos puedan acceder a él (Mayorga, 2017).

- **El adware** es un tipo de programa bastante polémico y difícil de catalogar. Algunos lo consideran una clase de spyware, mientras que otros aseguran que ni siquiera puede ser considerado un malware porque su intención final no es la de dañar los ordenadores principales. (Mayorga, 2017).
- **El ransomware:** Ransom quiere decir "rescate" en inglés, y de hecho lo que hace es secuestrar los datos de un ordenador y pedir un rescate económico a cambio de liberarlo. Normalmente lo que hace es cifrar tus datos, y lo que te ofrecen a cambio del rescate económico es la clave para poder descifrarlos. Este tipo de programas puede acceder a tu ordenador a lomos de un gusano informático u otro tipo de malware, y una vez cifre tus datos bloqueará tu ordenador mostrándote una pantalla de advertencia en la que se te informa que has sido víctima del ataque. En esa pantalla se te muestra también la cantidad a pagar y el método de pago, que puede ser por SMS, Paypal o mediante bitcoins (Mayorga, 2017).
- **Criptominería:** La Organización Kaspersky Lab (2019), explican que "por lo general, el malware de criptominería se instala en los equipos de los usuarios o de empresas junto con programas publicitarios, juegos hackeados y otro contenido pirateado" (p. 3). Al respecto, Mundaca (2020), agrega que "este malware apareció junto con la actividad de minería de criptomonedas, con lo cual se requiere de mucha capacidad de procesamiento. Este malware usa el computador infectado para minar criptomonedas sin que la víctima se dé cuenta" (párr.6).

Retomando los aportes de Mundaca (2020), se ha determinado se han explicado los Worms y los Botnets de la siguiente forma:

- **Worms:** Son verdaderos parásitos o gusanos computacionales. Es considerado una subclase de virus. Se propagan de ordenador a ordenador, sin necesidad de la intervención humana. Lo más peligroso de los worms es su capacidad para replicarse en tu sistema, por lo que podrías enviar, sin saberlo, cientos o miles de copias del gusano, creando un efecto devastador.
- **Botnets:** Red o grupo de ordenadores zombies, controlados por el propietario de los bots o computador zombie. El propietario de las redes de bots da instrucciones a los zombies. Estas órdenes pueden incluir la propia actualización del bot, la descarga de una nueva amenaza, el mostrar publicidad al usuario o el lanzar ataques de denegación de servicio, entre otras. En el último tiempo, se han utilizado este grupo de ordenadores para el minado de criptomonedas, sin que los usuarios legítimos superan qué sucedía (párr. 7-8).

5.6. Análisis del código fuente de un ransomware escrito en Python

En un análisis del código fuente de un ransomware escrito en Python bajo el nombre de detección dado por ***welivesecurity***: Python/Filecoder.AX. Siendo este una forma visible del código que ahora ya no está vigente, pero lo estuvo durante 2017 y 2018, siendo distribuida en un empaquetable ejecutable .exe a través de PyInstaller (figura 6).

Figura 6.

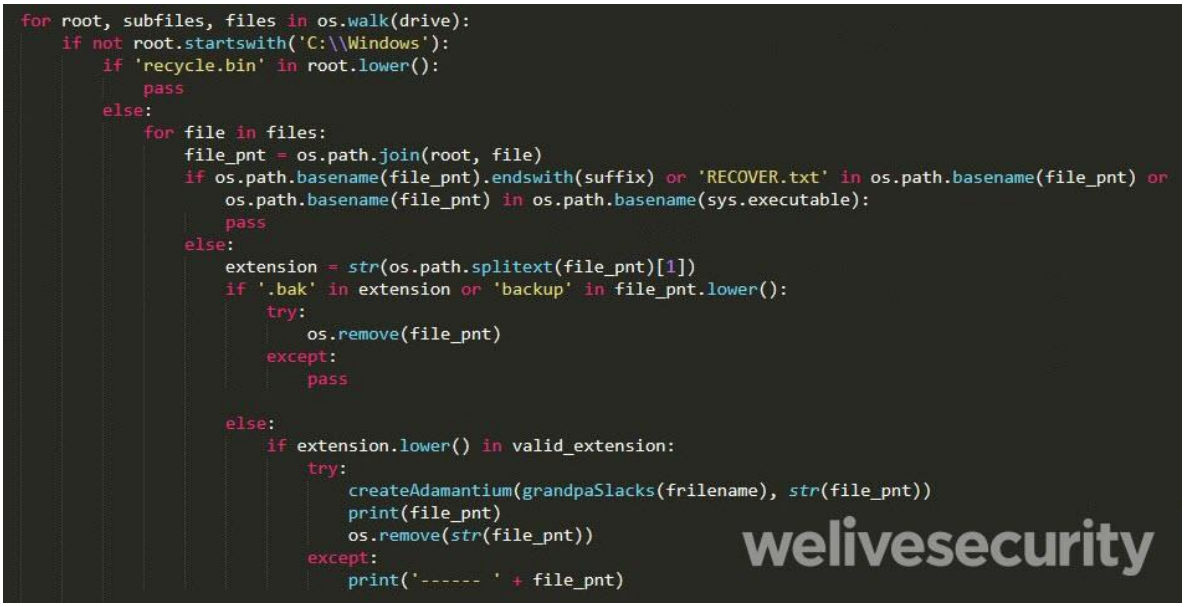
Análisis del Código Fuente al ejecutarse un Ransomware (parte 1).

Fuente: tomada de Kundro (2020).

Se ejecuta la función “run_crypt” para cada una de las unidades de almacenamiento presentes en la computadora, siendo “run_crypt” el componente principal de la amenaza. Además de llamar a dicha función, este código se encarga de generar un código para identificar la computadora de la víctima y que más adelante, luego de ser hasheado, será utilizado como clave para cifrar los archivos (figura 7).

Figura 7.

Análisis del Código Fuente al ejecutarse un Ransomware (parte 2).



Fuente: tomada de Kundro (2020).

Esta función realiza la misma acción repetidamente uno por uno sobre todos los archivos presentes en la unidad con algunas excepciones:

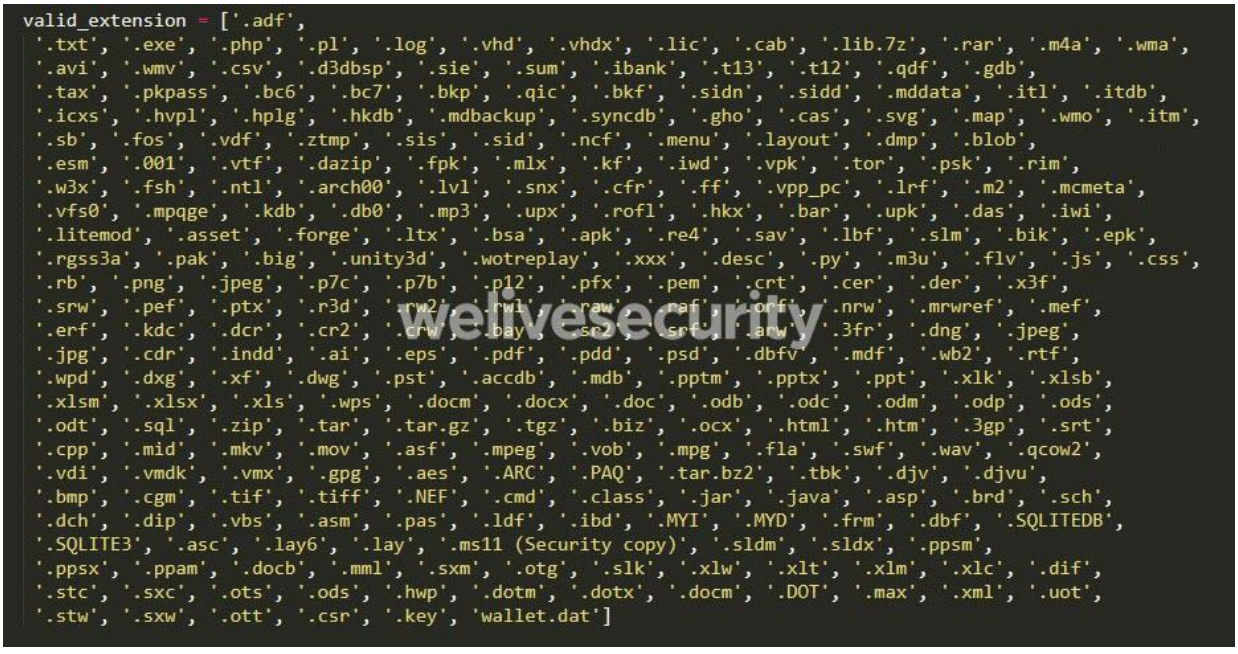
- Omite todos los archivos cuya ruta comience con “C:\Windows” es decir, no cifra los archivos del sistema para no afectar la estabilidad de este y permitir que la víctima pueda leer el mensaje de rescate.
- Omite todos los archivos que se encuentran en la papelera de reciclaje, lo cual tiene sentido dado que se supone que son archivos que la víctima no desea.
- Omite los archivos ya cifrados (.RQUILT[12]).

[12] Extensión de un archivo cifrado

- Omite los archivos donde escribe el mensaje de rescate para la víctima (RECOVER.txt).
- Omite el ejecutable de sí mismo.
- Omite los archivos .bak o aquellos cuyo nombre o ruta contenga la palabra “backup”. Estos son eliminados directamente.
- Omite los archivos que no tienen una extensión considerada valida o relevante para el ransomware (basándose en la lista contenida en la variable valid_extension).
- Luego, todos los archivos que pasen los controles previamente enunciados serán cifrados mediante la función “createAdamantium” y el archivo original no cifrado será eliminado inmediatamente.

Figura 8.

Análisis del Código Fuente al ejecutarse un Ransomware (parte 3).



```
valid_extension = ['.adf',
 '.txt', '.exe', '.php', '.pl', '.log', '.vhd', '.vhdx', '.lic', '.cab', '.lib.7z', '.rar', '.m4a', '.wma',
 '.avi', '.wmv', '.csv', '.d3dbsp', '.sie', '.sum', '.ibank', '.t13', '.t12', '.qdf', '.gdb',
 '.tax', '.pkpass', '.bc6', '.bc7', '.bkp', '.qic', '.bkf', '.sidn', '.sidd', '.mddata', '.itl', '.itdb',
 '.icxs', '.hvpl', '.hplg', '.hkdb', '.mdbackup', '.syncdb', '.gho', '.cas', '.svg', '.map', '.wmo', '.itm',
 '.sb', '.fos', '.vdf', '.ztmp', '.sis', '.sid', '.ncf', '.menu', '.layout', '.dmp', '.blob',
 '.esm', '.001', '.vtf', '.dazip', '.fpk', '.mlx', '.kf', '.iwd', '.vpk', '.tor', '.psk', '.rim',
 '.w3x', '.fsh', '.ntl', '.arch00', '.lvl', '.snx', '.cfr', '.ff', '.vpp_pc', '.lrf', '.m2', '.mcmeta',
 '.vfs0', '.mpqge', '.kdb', '.db0', '.mp3', '.upx', '.rofl', '.hkx', '.bar', '.upk', '.das', '.iwi',
 '.litemod', '.asset', '.forge', '.ltx', '.bsa', '.apk', '.re4', '.sav', '.lbf', '.slm', '.bik', '.epk',
 '.rgss3a', '.pak', '.big', '.unity3d', '.wotreplay', '.xxx', '.desc', '.py', '.m3u', '.flv', '.js', '.css',
 '.rb', '.png', '.jpeg', '.p7c', '.p7b', '.p12', '.pfx', '.pem', '.crt', '.cer', '.der', '.x3f',
 '.srw', '.pef', '.ptx', '.r3d', '.rw2', '.rwl', '.raw', '.raf', '.orf', '.nrw', '.mrwref', '.mef',
 '.erf', '.kdc', '.dcr', '.cr2', '.crw', '.bay', '.sr2', '.srf', '.arw', '.3fr', '.dng', '.jpeg',
 '.jpg', '.cdr', '.indd', '.ai', '.eps', '.pdf', '.pdd', '.psd', '.dbfv', '.mdf', '.wb2', '.rtf',
 '.wpd', '.dxg', '.xf', '.dwg', '.pst', '.accdb', '.mdb', '.pptm', '.pptx', '.ppt', '.xlk', '.xlsb',
 '.xlsm', '.xlsx', '.xls', '.wps', '.docm', '.docx', '.doc', '.odb', '.odc', '.odm', '.odp', '.ods',
 '.odt', '.sql', '.zip', '.tar', '.tar.gz', '.tgz', '.biz', '.ocx', '.html', '.htm', '.3gp', '.srt',
 '.cpp', '.mid', '.mkv', '.mov', '.asf', '.mpeg', '.vob', '.mpg', '.fla', '.swf', '.wav', '.qcow2',
 '.vdi', '.vmdk', '.vmx', '.gpg', '.aes', '.ARC', '.PAQ', '.tar.bz2', '.tbk', '.djv', '.djvu',
 '.bmp', '.cgm', '.tif', '.tiff', '.NEF', '.cmd', '.class', '.jar', '.java', '.asp', '.brd', '.sch',
 '.dch', '.dip', '.vbs', '.asm', '.pas', '.ldf', '.ibd', '.MYI', '.MYD', '.frm', '.dbf', '.SQLITEDB',
 '.SQLITE3', '.asc', '.lay6', '.lay', '.ms11 (Security copy)', '.sldm', '.sldx', '.ppsm',
 '.ppsx', '.ppam', '.docb', '.mml', '.sxm', '.otg', '.slk', '.xlw', '.xlt', '.xlm', '.xlc', '.dif',
 '.stc', '.sxc', '.ots', '.ods', '.hwp', '.dotm', '.dotx', '.docm', '.DOT', '.max', '.xml', '.uot',
 '.stw', '.sxw', '.ott', '.csr', '.key', 'wallet.dat']
```

Fuente: tomada de Kundro (2020).

Se pueden observar tres elementos interesantes: (Variable "valid_extension", Función "createAdamantium", Función "grandpaSlacks"). Esta variable contiene una lista donde se definen todas las extensiones de archivo que serán consideradas validas o relevantes y, por ende, que serán cifradas por este ransomware, como se ve en la Figura 8.

5.7. Inteligencia Artificial

Consiste en la simulación de procesos que realiza la naturaleza humana por biología en máquinas, es decir, se emplean sistemas informáticos para imitar las funciones cognitivas fundamentales, como lo son el aprendizaje, la autocorrección y el razonamiento. Para llevar a cabo estas tres últimas funciones se emplea la combinación de algoritmos Actualmente se pueden encontrar muchos ejemplos de tecnología de IA como procesamiento del lenguaje natural, robótica, automatización, visión por computador y Machine Learning (Zufiaurre, 2019). Esta última es de la que se hará uso en este proyecto.

5.8. Machine Learning

El aprendizaje máquina, en inglés, Machine Learning es una disciplina de la Inteligencia Artificial. Esta disciplina es capaz de crear sistemas que pueden aprender automáticamente basándose en millones de datos, identificar patrones y tomar decisiones con mínima intervención humana. El sistema informático que se crea revisa datos y es capaz de predecir comportamientos futuros, es decir, mediante una serie de instrucciones representa la solución a un problema. Su objetivo principal es que las máquinas sean capaces de aprender como un humano lo haría (Zufiaurre, 2019).

El principal objetivo va enfocado a que los sistemas puedan tener la capacidad de generar y asociar, en casos familiares como en nuevos o imprevistos. Esto es posible mediante la formación de modelos que generalicen la información que se les presenta para realizar sus predicciones. La información a la que se enfrenta es lo que se conoce como Big Data. (Zufiaurre, 2019).

El rápido crecimiento de las fuentes de información digital hace imposible que un humano sea capaz de procesar y analizar tal cantidad de datos, por lo tanto, técnicas como el aprendizaje de máquina se ha popularizado por la variedad de modelos con los que se puede analizar grandes volúmenes de datos de forma más rápida y precisa. Entre esos modelos están: las redes neuronales, las máquinas de soporte vectorial, los algoritmos de agrupamiento como los KNN (K vecinos más cercanos), la regresión logística, el aprendizaje simbólico e introducción de reglas, algoritmos basados en evolución, el aprendizaje analítico y métodos híbridos, entre otros.

5.9. Técnicas de Machine-Learning

- **Aprendizaje supervisado:** Este tipo de aprendizaje involucra alta participación humana en términos de monitoreo, selección de nuevos atributos, entrenamiento y nuevas puestas en producción. El aprendizaje supervisado utiliza un set de datos que es dividido en 2 partes: una de entrenamiento, para generalizar el conocimiento que se espera pueda predecir el modelo resultante y los datos restantes se toman como test y se usan para verificar y contrastar que tan bien aprendió a generalizar el conocimiento el modelo final. Las más usuales en el aprendizaje supervisado están los Árboles de decisión, Máxima entropía, Naives Bayes y Support vector machine.

- **Aprendizaje no supervisado:** Esta clase de algoritmos busca identificar estructuras en los datos. No se tiene la respuesta conocida para cada caso por lo que el algoritmo debe encontrar las relaciones entre las variables involucradas. No se busca la representación de los datos. Las más usuales en las no supervisadas son las técnicas de clustering.
- **Aprendizaje semi supervisado:** Estos algoritmos son un hibrido de aprendizaje supervisado y no supervisado. En este caso, el algoritmo trabaja con pequeñas cantidades de datos de entrenamiento etiquetados y más de datos sin etiquetar. De este modo, se hace uso creativo de los métodos supervisados y no supervisados para resolver una tarea determinada. En las semi-supervisadas tenemos lasTécnicas transductive support vector machine y Expectation maximizatión
- **Aprendizaje por refuerzo:** El Algoritmo recibe algún tipo de valoración acerca de la idoneidad de la respuesta dada. Ejemplo: darle una orden al perro de sentarse, si lo hace se le premia y si no lo hace se le castiga.
- **Aprendizaje en lotes:** También se denomina como aprendizaje fuera de línea. Este tipo de aprendizaje es utilizado cuando se tiene un conjunto de datos de entrada y se quiere correlacionar con un conjunto de datos de salida, es decir encontrar una conexión entre estos conjuntos, que en la mayoría de los casos son conjuntos de datos diferentes por lo que deben ser normalizados para obtener el resultado. Ejemplo: la correlación que existe entre la estatura y el peso de una persona
- **Aprendizaje en línea:** En este caso el aprendizaje no se detiene una vez que los datos están disponibles, sino que los datos se introducen en el sistema en mini-lotes y el proceso de aprendizaje continúa con nuevos lotes de datos.

6. Desarrollo del Proyecto

De conformidad con los propósitos de la presente investigación, en esta sección se desarrollan cada uno de los objetivos específicos formulados en los apartados anteriores; iniciando con la identificación de las técnicas de aprendizaje automático más empleadas en años recientes para la detección de software malicioso, haciendo precisión en el malware ransomware, de acuerdo con el sondeo bibliográfico efectuado para este fin. Después se presenta un comparativo de los métodos basados en Machine Learning más representativos encontrados en la literatura, de acuerdo con las métricas de eficacia en la detección y el tipo de malware para el cual se diseñan; también se caracteriza el ransomware como objeto de estudio. Se finaliza este capítulo con la propuesta de la técnica de detección de malware-ransomware, a partir de su caracterización y de los modelos de aprendizaje automático, identificados y analizados con los primeros objetivos específicos.

6.1. Identificación de las técnicas de Machine-Learning para la detección de Malware–Ransomware

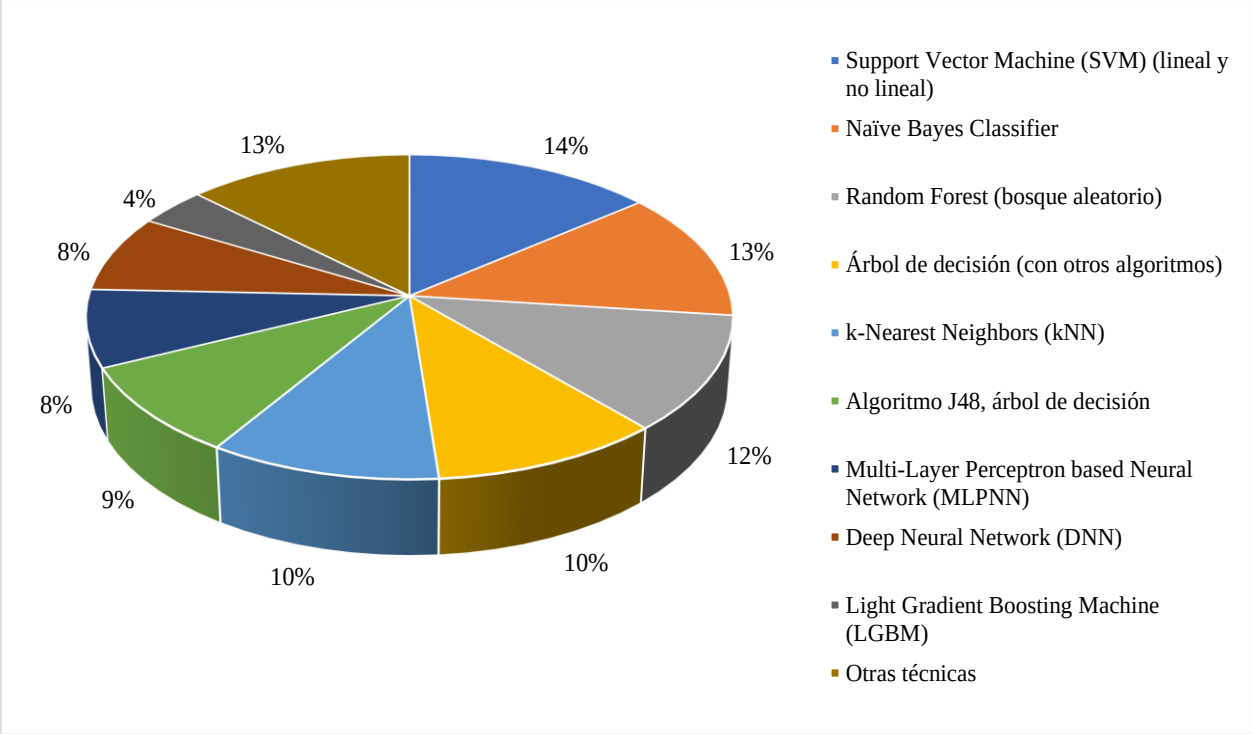
Para efectos de este proyecto, se realizó la revisión y análisis de 35 documentos, tanto artículos científicos hallados en diversas bases de datos digitales, como trabajos de grado, nacionales e internacionales, con una temporalidad comprendida entre 2016 y 2021; los cuales dieron cuenta de las diferentes técnicas de aprendizaje automático aplicadas y evaluadas en el ámbito investigativo, para la clasificación y detección de diversos tipos de software malicioso, precisando que en algunos casos dicha valoración se efectuó explícitamente con respecto a determinados ransomware, tal como se indica en acápites posteriores.

En la revisión fue posible confirmar que el malware continúa siendo una amenaza para los sistemas de información domésticos y empresariales, debido al surgimiento cada vez, de nuevos tipos de software malicioso más infecciosos, potentes y que generan mayores daños para los afectados. Ante esta realidad, se evidencia el estudio y desarrollo de múltiples técnicas y modelos que contribuyen con la detección y la predicción del comportamiento que tienen estos programas maliciosos, encontrándose que efectivamente las técnicas de Machine Learning basadas en Inteligencia Artificial favorecen estos procesos.

En los estudios consultados y analizados se identificó que los experimentos para la detección y clasificación de datos malware se realizan con mayor frecuencia usando las técnicas de Machine Learning Máquinas de Vectores Soportes (SVM, lineal y no lineal) (14%) y el Clasificador Naïve Bayes, con una participación porcentual igual a 13%. Se tienen, además, los métodos Random Forest (bosque aleatorio) con una proporción de 12%, seguido del Árbol de Decisión (10%), teniendo en cuenta que, con esta última técnica, se utiliza principalmente, el algoritmo J48 que por sí mismo ocupa un 9%. Se aplican también los modelos de aprendizaje automático: k vecinos más cercanos (kNN) (10%), Perceptrón Multicapa Basado en una Red Neuronal (MLPNN) (8%), las Redes Neuronales Profundas (DNN) con una frecuencia del 8% y la Máquina con Aumento de Gradiente de Luz (LGBM), igual a una proporción del 4%. Otros algoritmos aparecen en menor porcentaje sumando como grupo un 13%; con respecto a estas otras técnicas de aprendizaje automático halladas durante la revisión bibliográfica y que arrojaron las menores puntuaciones porcentuales, se encuentran: Regresión Logística (2,6%), Deep Belief Network (DBN) (2,6%), Random Tree (Árbol Aleatorio) (2,6%), Algoritmo AdaBoost (2,6%), Bagging (1,4%) y File Large Margin (FLM) (1,4%). Esta distribución se puede ver en la figura 9.

Figura 9.

Frecuencia de uso de las técnicas de Machine Learning identificadas en la literatura



Fuente: elaboración propia.

Con los hallazgos de la figura 9, se confirman los resultados obtenidos por Godoy (2017), ya que a través de su investigación identificó, desde la literatura, las principales técnicas de aprendizaje de maquina utilizadas en minería de texto para procesar, extraer y resumir información útil de grandes cantidades de fuentes documentales. En este caso, se lleva a cabo un estudio documental mediante la revisión de diversos artículos en ingles de revistas indexadas en diferentes bases de datos en internet. Se hallaron 56 artículos al respecto, correspondientes a 162 autores en total y se hallaron 13 técnicas de aprendizaje automático aplicadas anteriormente, en minería de texto; encontrándose que: la técnica mencionada con mayor frecuencia en la literatura es Support

Vector Machine (SVM) (16,88%), de igual forma, esta técnica es usada en un 20,59% en forma comparativa con otros métodos y también como técnica principal con una frecuencia de 22,72% y la menos representativa es la técnica Multinomial Naive Bayes (MNB). Esto hace evidente la flexibilidad en la aplicación que tienen los algoritmos de Machine Learning para las tareas de clasificación y agrupamiento; también que los algoritmos SVM y MNB son los más trabajados por su estabilidad y buena respuesta de evaluación.

Las técnicas referenciadas suelen ser evaluadas por lo general, a través de las métricas de eficacia: precisión y exactitud, con la finalidad de determinar la tasa de predicciones correctas efectuadas por el modelo y la proporción entre los positivos reales y todos los casos positivos. Adicionalmente, también se resaltan métricas como: el tiempo de creación del modelo, el área bajo la curva ROC (AUC), la tasa de verdaderos positivos y de falsos positivos. Los resultados obtenidos con estas métricas permiten establecer que modelos de clasificación exhiben las mejores puntuaciones, características y más alto rendimiento. Según los documentos revisados son: Random Forest (bosque aleatorio), Naïve Bayes, árbol de decisión con el algoritmo J48 y SVM; resultados que dependen directamente de la calidad de los datos y los procedimientos de preparación efectuados con los datos, ya que se identifican diferencias entre el rendimiento de las técnicas utilizadas cuando son datos sin procesar y cuando son procesados e integrados.

Sumado a esto, se encuentra con las investigaciones analizadas que con frecuencia sus autores aprovechan las ventajas que ofrecen diversas plataformas de software en línea, muchas de ellas gratuitas, las cuales facilitan la ejecución de cada una de las etapas que comprenden los experimentos de detección y clasificación de los modelos propuestos; herramientas que apoyan estas actividades, desde la preparación y el procesamiento de los datos hasta la ejecución de las técnicas y modelos seleccionados e incluso, contribuyen con el robustecimiento de los modelos

propuestos mediante la integración y el ensamble de varias técnicas de aprendizaje automático. Así mismo, se hallaron siete estudios con los cuales se utilizaron los archivos de malware informático del sistema operativo Android.

En materia de ransomware, son limitadas las investigaciones enfocadas específicamente en la detección y clasificación de este tipo de malware, por lo general orientadas en la aplicación de alguna de las técnicas mencionadas en líneas previas, pero también se observa el interés de los investigadores por validar otras herramientas y modelos para llevar a cabo estas acciones, es el caso del Reconocimiento Óptico de Caracteres (OCR), el volcado de la memoria RAM y la extracción de archivos relevantes, información que se explica posteriormente. Particularmente, el Reconocimiento Óptico de Caracteres (OCR) es una de las técnicas utilizadas para la detección y clasificación de malware ransomware, la cual, según los datos anteriores, se integra a la categoría de Perceptrón Multicapa Basado en una Red Neuronal (MLPNN), porque el OCR es una técnica de aprendizaje profundo, de la familia de redes neuronales convolucionales.

Expuesto lo anterior, en las siguientes líneas de detallan los procedimientos y aplicaciones efectuados por los autores con relación a las técnicas más usadas en la literatura, incluyendo la explicación de aquellos estudios enfocados explícitamente en el malware ransomware. Es preciso clasificar que en ciertas investigaciones se ha aplicado de forma unitaria alguna de las técnicas identificadas durante el desarrollo de este proyecto; aunque en la mayoría de estos estudios, se han utilizado múltiples métodos, por lo general, con la pretensión de comparar los resultados arrojados con cada una de estos y determinar en función de una serie de métricas de evaluación, la mejor alternativa. Por lo tanto, la ubicación de la información en el caso de los estudios con varias técnicas se realizó según las técnicas con mejores resultados en cuanto a su evaluación.

6.1.1. Support Vector Machine (SVM) (lineal y no lineal: kernelizado).

Se validó que una de las técnicas de aprendizaje automático más utilizada para la detección y clasificación de malware es la Máquina de Vectores Soportes (SVM), la cual ha sido utilizada con diversas aplicaciones. Las Máquinas de Vectores de Soporte (Support Vector Machines) permiten encontrar la forma óptima de clasificar entre varias clases, la clasificación óptima se realiza maximizando el margen de separación entre las clases y los vectores que definen el borde de esta separación son los vectores de soporte. En el caso de que las clases no sean linealmente separables, se utiliza el truco del kernel para añadir una o más dimensiones nuevas donde sí lo sean. Las máquinas de vectores de soporte son muy utilizadas porque la propiedad de obtener el margen de separación máximo es muy atractiva, sobre todo en aplicaciones como detección de malware donde un falso negativo puede resultar fatal.

En este sentido, Nivaashini et al. (2018), realizan un análisis experimental dinámico de los malware del sistema operativo Android, a partir de los permisos del documento de manifiesto habilitados para cada aplicación y mediante la detección y clasificación comparativa de técnicas de Machine Learning utilizando Weka[13]. La base de datos está conformada por 2.999 aplicaciones ordinarias y 1.999 Apps de malware, la extracción de los atributos se efectúa con el archivo APK y su selección se realiza con la ejecución de: 1) evaluador de atributos de relación de ganancia, 2) evaluador de atributos de alivio, 3) evaluador de subconjuntos de características basado en correlación (CFS) y 4) el análisis chi-cuadrado (CH); hallándose que el análisis de chi-cuadrado produce los 34 atributos más significativos. En el estudio se evalúan seis técnicas de Machine Learning ejecutadas con Weka, a saber: Naïve Bayes (NB), J48, Random Forest (RF), Support

[13] Weka es una plataforma GNU de software destinada al aprendizaje automático y la minería de datos.

Vector Machine (SVM), Multi-Layer Perceptron based Neural Network (MLPNN), k-Nemost Neighbor (kNN); esto teniendo en cuenta las siguientes métricas: Exactitud (ACC), Tasa de verdaderos positivos (TPR), Tasa de falsos positivos (FPR) y Precisión. Se encontró que la técnica SVM arroja el mayor nivel de rendimiento que otros algoritmos de clasificación, cuando se combina con la técnica de selección de atributos del análisis de chi-cuadrado y el clasificador Naïve Bayes mostró mejores resultados en la categorización del conjunto de datos de malware, conforme con su complejidad computacional.

Ahora bien, Uchnár y Feciľak (2019), se plantearon como objetivo principal, la comparación de algoritmos de aprendizaje automático con el fin de analizar el comportamiento del malware; para lo cual se recopilaron 1000 archivos en total, 750 archivos malignos obtenidos de diversos honeypots a través de Internet y 250 archivos benignos principalmente, del sistema Windows. Posteriormente, se recopilaron datos de análisis de comportamiento de dichos archivos, utilizando la herramienta Cuckoo Sandbox, ya que es una de las herramientas de código abierto más populares para el análisis del comportamiento de malware y puede proporcionar informes exhaustivos de archivos. Luego se analizó cada archivo, con la restricción de 120 segundos de ejecución para evitar bucles infinitos, y se recopilaron los informes.

Adicionalmente, Uchnár y Feciľak (2019), dividieron los datos en dos grupos, así: de entrenamiento 550 archivos malignos y 150 benignos y de prueba 200 malignos y 100 benignos; y se seleccionaron los algoritmos de Machine Learning para la comparación, a saber: K-Nearest Neighbors (kNN), Naive Bayes y Random Forest, con los cuales se efectuó un proceso de validación cruzada en dos niveles: Durante la fase de aprendizaje y como como una nueva validación de predicción de muestras. Los indicadores calculados fueron: Verdaderos positivos, negativos verdaderos, falsos positivos y falsos negativos; y las métricas de evaluación

comparativa, fueron: Precisión, exactitud, sensibilidad y Puntuación F1. Se concluye que el algoritmo Random Forest obtuvo los mejores resultados con un 96% de recuperación.

Por otra parte, García (2020) realizó una aplicación para dotar al usuario y que este defienda su información del malware. La aplicación le indicará los procesos que se estén ejecutando actualmente en su sistema; del mismo modo, le permite ejecutar un análisis de la red en cuanto a tamaño, puerto y procedencia del paquete que se esté iniciando. La metodología aplicada fue SCRUM y las herramientas utilizadas, fueron: Nessus, Nmap, Open Vas, Yara, Librería PCAP y SVM, creando una función, Upnp Devices y Sniffer. El autor recomienda para trabajos futuros integrar redes neuronales convolucionales dentro de los clasificadores de Machine Learning.

6.1.2. Naive Bayes Classifier (NBC)

Los modelos de Naive Bayes son una clase especial de algoritmos de clasificación de Aprendizaje Automático, se basan en una técnica de clasificación estadística llamada "teorema de Bayes" donde se asume que las variables predictoras son independientes entre sí. Proporcionan una manera fácil de construir modelos con un comportamiento muy bueno debido a su simplicidad y lo consiguen proporcionando una forma de calcular la probabilidad 'posterior' de que ocurra un cierto evento, dadas algunas probabilidades de eventos 'anteriores'. Esta propiedad y facilidad de implementación es aprovechada sobre todo en etapas de selección de características y clasificación.

En el artículo publicado por Kim et al. (2020), cuyo propósito es detectar empacadores personalizados y conocidos con características efectivas, mediante un experimento comparativo extenso entre algoritmos de aprendizaje automático junto con la utilización de la entropía para la extracción de un total de 13 funciones de todo el malware y del encabezado del archivo ejecutable

portátil (PE) para detectar si están empaquetados o no. La clasificación de los archivos de malware basados en PE configurados en empaquetados o no, se efectúa a través de ocho modelos de aprendizaje automático, a saber: árbol de decisión, bosque aleatorio, SVM lineal, SVM no lineal, kNN, NBC, regresión logística y perceptrón multicapa; siendo evaluados con las métricas de precisión y puntuación F[14]. Los mejores resultados en la detección de empaquetamiento los arrojaron las técnicas de Naive Bayes, SVM (no lineal) y árbol de decisión; el modelo kNN muestra el rendimiento más bajo y el perceptrón multicapa puede mejorar sus resultados de detección con la construcción de una red neuronal más amplia y profunda.

Choudhary y Sharma (2020) desarrollan un estudio enfocado en identificar malware a partir de su detección y clasificación mediante la aplicación de los cálculos de aprendizaje automático, comparando un modelo de clasificación multinomial Naive Bayes con un método de firmas de Bytes tradicional. Los cálculos obtenidos con este experimento revelan que el categorizador Multinomial Naive Bayes tuvo la tasa de detección más elevada, y el método de firmas tuvo la menor tasa de falsos positivos. En general, la precisión más alta es la calculada con el Clasificador Naive Bayes, una mezcla de métodos, sin embargo, puede tener resultados generales más satisfactorios.

[14] La puntuación F en una métrica que relaciona la precisión y la sensibilidad, en otras palabras, la proporción FP versus FN.

6.1.3. Random Forest (bosque aleatorio)

Un Random Forest es un conjunto (ensamble) de árboles de decisión combinados con bagging[15]. Lo que en realidad está pasando, es que distintos árboles ven distintas porciones de los datos; ningún árbol ve todos los datos de entrenamiento, esto hace que cada árbol se entrene con distintas muestras de datos para un mismo problema y de esta forma, al combinar sus resultados, unos errores se compensan con otros y se logra una predicción que generaliza mejor la clasificación. En conjuntos de características muy grande como puede ser el caso de los conjuntos de datos con muestras de atributos de malware, trabajar con Random Forest reduce el tiempo de entrenamiento y mejora los tiempos de respuesta; sin embargo, al trabajar con subgrupos (problema fraccionado) los árboles de decisión tienen la tendencia de sobre-ajustar (overfit); esto quiere decir que tienden a aprender muy bien sus datos de entrenamiento pero su generalización no es tan buena, sobre todo con muestras no conocidas en el entrenamiento.

Con respecto a la aplicación de la técnica Random Forest, el estudio desarrollado por Radwan (2019), quien se enfoca en la clasificación con alta precisión de un archivo Ejecutable Portátil (PE) como benigno o malware, mediante un método de análisis estático para extraer el conjunto de características integradas utilizado en la implementación de la solución propuesta y que son seleccionadas de los tres encabezados principales de los archivos PE y un conjunto de características derivadas, esto con la finalidad de mejorar el rendimiento de la predicción y la eficiencia del modelo. Se crea una función integrada con un total de 74 características, un conjunto conformado por las 28 características sin procesar reducidas (RRF) seleccionadas, 26

[15] Método para combinar varios modelos de Machine Learning que se entrenan con subconjuntos que se forman eligiendo muestras aleatoriamente (con repetición) del conjunto de entrenamiento, y cuyo resultado se combina utilizando la votación por mayoría, con un voto suave para los modelos que den probabilidades más bajas.

características sin procesar expandidas y 20 características derivadas existentes en el conjunto de características integradas (IntF). Adicionalmente, se utiliza el método de división de prueba de tren (70/30) y la validación cruzada de 10 veces (CV) para la división del conjuntos sin procesar e integrados y para evaluar los algoritmos de clasificación Machine Learning, teniendo en cuenta que se eligieron las siete técnicas con mejores capacidades de generalización que son: K-Vecinos más cercanos (KNN), Árboles reforzados con gradiente (GPR), Árbol de decisión (DT), Bosque aleatorio (RF), Margen grande de archivo (FLM), Regresión logística (LR) y Naïve Bayes (NB). Se usan la exactitud, precisión y medida f como métricas de rendimiento de cada clasificador, mediante la ejecución de los experimentos con el programa Windows 10 instalado en la máquina, utilizando un procesador Intel Core i7 Duo 2.6 GHz con una estación de trabajo de 8GB RAM. Los resultados obtenidos con las métricas evaluadas permiten determinar que todos los clasificadores evaluados con estas métricas, excepto KNN arrojan mejores resultados con el conjunto de datos IntF en comparación con el conjunto de datos sin procesar. En definitiva, se concluye que la técnica Random Forest muestra entre todos los clasificadores analizados, el rendimiento más alto, tanto con el conjunto de datos sin procesar como integrados.

Por otra parte, la organización Management Solutions (2018), mediante un ejercicio cuantitativo aplicado a un caso de estudio del sector financiero, con la finalidad de mejorar la eficacia de la modelización y adecuar los datos en relación con los algoritmos implementados realiza un comparativo con técnicas tradicionales, para ello se construye un modelo logístico y, además, se aplica una red elástica como técnica de regularización, se desarrollan dos métodos de ensamble: Random Forest y un Adaboost; así mismo, se estiman dos modelos SVM, uno con una función lineal y otro radial. Se calcula la medida de poder discriminante con una matriz de confusión fijando un punto de corte en el valor que minimiza el error de la predicción y el área

bajo la curva ROC (AUC) para evaluar las técnicas utilizadas; hallándose que el modelo Random Forest, con 80 variables y 50 árboles de decisión, arrojó los mejores resultados que los demás métodos, seguido de la red elástica; aquel se destacó en términos de tasa de aciertos, poder discriminante y área bajo la curva ROC.

Buscando comparar métodos, Stiawan et al. (2020), implementan cuatro algoritmos de aprendizaje automático mediante la clasificación de malware Botnet del IoT (Las redes de bots son una de las amenazas a la seguridad de las redes de Internet). Usan la herramienta Weka y Scikit-learn[16] Machine Learning de aprendizaje automático y se afianzan desde la literatura con varios experimentos mostrados allí por varios autores que afirman que el aprendizaje automático puede utilizarse para el proceso de clasificación de malware de botnets, luego se implementan con los algoritmos de AdaBoost, árbol de decisión, Random Forest y Naïve Bayes. Realizan experimentos para medir sus rendimientos en términos de precisión, tiempo de ejecución y tasa de falsos positivos. Los resultados de los experimentos muestran que la herramienta Weka proporciona métodos de clasificación más eficientes, mientras que, en la tasa de falsos positivos y negativos, el uso de Scikit-learn proporciona mejores resultados. El conjunto de datos N_BaIoT utilizado en este estudio tiene una distribución desequilibrada de clases de tráfico y ataques capturados de un banco de pruebas de tráfico de red (Dos timbres, termostatos, monitores, cámaras de seguridad y web) el conjunto de datos consiste en tráfico normal y tráfico de ataque que contienen 115 características, incluyendo 5 canales principales (canal, host-MAC e IP, networkjitter, host-IP y socket). Concluyen de los resultados experimentales que los árboles de decisión son los que mejor resultado dan en estos escenarios de prueba, que las herramientas Weka

[16] Es un módulo de Python para aprendizaje automático construido sobre SciPy y se distribuye bajo la licencia BSD de 3 cláusulas.

y Scikit-learn tienen ventajas en la rápida construcción y entrenamiento de los modelos y sus desventajas están en no contar con una amplia flexibilidad más allá del ajuste de hiperparametros.

Ahora bien, con la investigación de Darus et al. (2018), se analiza el malware informático dentro del SO Android con la aplicación de una técnica basada en la visualización de imágenes y con la utilización de 483 imágenes en escala de grises, correspondientes a 183 archivos APK malware y 300 benignos; los cuales fueron descomprimidos para extraer el archivo classes.dex, que es el archivo binario que contiene el código de operación de Dalvik. La extracción de las características de las imágenes se realizó con un GIST descriptor[17] y para la clasificación se usaron tres algoritmos de aprendizaje automático diferentes: k vecino más cercano (KNN), Random Forest (RF) y Árbol de decisión (DT). Se identificó que con el uso del algoritmo de aprendizaje automático de Random Forest se obtuvo el mayor porcentaje de precisión de detección.

6.1.4. Árbol de decisión, con énfasis en el algoritmo J48

J48 es una implementación open source en lenguaje de programación Java del algoritmo C4.5 en la herramienta Weka de minería de datos. C4.5 es un algoritmo desarrollado por Ross Quinlan y usado para generar árboles de decisión para clasificación estadística, basado en la minimización de la entropía. La entropía de información de categoría representa la suma de las incertidumbres de varias categorías en todas las muestras; según el concepto de entropía, cuanto mayor es la entropía, mayor es la incertidumbre y se necesita más información para resolver la

[17] GIST Descriptor, traduce "descriptor esencial" y hace referencia a una función de imagen que proporciona una conexión entre la información visual y semántica; la cual en este caso, puede ser utilizado como base para una puntuación eficaz de malware basada en imágenes.

clasificación. En este sentido el algoritmo J48 permite un uso eficiente de los atributos de entrada para optimizar el árbol de decisión.

Se identificó la publicación de Rodríguez (2018), quien se propuso en demostrar la utilidad y aplicaciones que se pueden realizar con Machine Learning para la detección de conexiones maliciosas, seleccionando el mejor algoritmo al respecto, de acuerdo con la máxima precisión obtenida para la identificación de ataques. En este caso, la evaluación de sistemas de detección de anomalías; utilizó el 10% del dataset KDDCUP'99, 10 atributos y la herramienta Weka para la elección del mejor algoritmo de Machine Learning, mediante la evaluación de su rendimiento, según el tiempo de creación del modelo y la precisión de acuerdo con la confianza del porcentaje de instancias clasificadas correctamente. Se valoraron las técnicas: Random Tree, Random Forest, J48, Naïve Bayes, SVM y regresión logística. La aplicación acogida fue la enfocada en el entrenamiento de un árbol de decisión con el algoritmo J48, dado que esta arrojó las puntuaciones más altas con relación a las métricas evaluadas, las mismas que fueron mejoradas con la eliminación de la normalización y aumentando el tamaño del Random Tree.

Adicionalmente, Moscardó (2018), se interesó en diseñar y desarrollar un detector predictivo de ataques web, accesible vía webservice; utilizando un conjunto de datos CSIC 2010 constituido por ataques HTTP de 3 tipos: estáticos, dinámicos y solicitudes ilegales involuntarias, y a su vez convertido a CSV para su uso con Weka y Scikit-learn. La definición del algoritmo de aprendizaje supervisado de Machine Learning utilizado para esta propuesta, se realizó mediante a evaluación del grado de desempeño, según las siguientes métricas: tiempo de construcción del modelo, tiempo de validación del conjunto de prueba y exactitud (tasa de aciertos). Los principales resultados obtenidos, son: el árbol de decisión J48 con el mejor porcentaje de clasificaciones correctas, K-Nearest Neighbours es el algoritmo que genera el modelo más rápidamente y el J48

es el más rápido en cuanto al tiempo empleado para verificar el modelo con el conjunto de datos test. Se determina que el algoritmo que mejor se adapta a la investigación es el árbol de decisión J48, implementado con la librería Scikit-learn, se entrena con el 58% de instancias y se evalúa su exactitud y predicción con el 15% de instancias restantes. Con este modelo predictivo se construye el webservice detector supervisado de amenazas, el cual actúa permitiendo o bloqueando su conexión en la red.

En este mismo sentido, Al-Janabi y Altamimi (2020), proporcionaron un estudio que determina los mejores métodos de extracción y clasificación de características y que den como resultado la mejor precisión en la detección de malware mediante el aprendizaje automático. Inician analizando las diferentes investigaciones de métodos de detección mencionadas por diferentes autores desde la literatura, y se han ido clasificado en función de su técnica de análisis, ya sea estática, dinámica o híbrida. Se observan sus resultados y se concluye que el algoritmo J48 y el análisis híbrido superaron a los demás métodos con una precisión del 100% en la detección de malware en el sistema Windows y por otro lado, se ha conseguido la misma precisión en el sistema Android al emplear el algoritmo de árbol de decisiones mediante el análisis dinámico. Cabe aclarar que la precisión de los modelos propuestos varía en función del método utilizado, el número de atributos, el conjunto de datos y los registros del conjunto de datos, las técnicas de preprocesamiento y las herramientas implementadas en el modelo. Las demás técnicas que fueron comparadas son: Inductive Rule Learner (RIPPER, Aprendizaje de Reglas Intuitivas, en español), Naives Bayes (NB), Multi-Naives Bayes, Gradient Boosting Machine (GBM, Máquina de Potenciación del Gradiente, en español), Red Neuronal de Sensibilidad (SNN), Random Forest (RF), k-Nearest Neighbors (kNN, k-Vecinos más Próximos, en español), Potenciación del Gradiente (GB), SVM-poly (SP, Máquina Vector Soporte con Kernel Polinomial, en español), DT,

IMDS, SVM, K-Means (K-Medias), Regresión Logística, Histogramas, Multilayer Perceptron Neural Network, Instance-Based Learner (IBK, Aprendizaje Basado en Instancias, en español), algoritmo J48 DT, J48-Graft (extractor de reglas y J48), Fuzzy Unordered Rule Induction Algorithm (FURIA, Algoritmo de inducción de reglas desordenadas difusas, en español).

En correspondencia con lo planteado en breve, Romero (2019), en esta oportunidad se trae a colación una investigación orientada hacia el análisis de técnicas de aprendizaje automático que permitan el diseño y creación de un clasificador de flujos en internet fiable acorde con la realidad, a través de la plataforma de software Weka. En este caso, el procesado y filtrado de los datos de los flujos capturados y cedidos por la Universidad Politécnica de Cataluña (UPC), se efectúa organizándolos según los 16 atributos determinados, convertidos en un archivo con el formato Atribute-Relation File Format, aceptado por Weka. Se continúa con la parametrización del software Weka, utilizando el algoritmo de árboles de clasificación J48 y se realizan diversos filtrados de los datos para calcular el porcentaje de clasificación alcanzado, hallándose que la combinación de conjuntos de datos mejora dicho porcentaje, la misma que se utiliza para construir un clasificador que detecte tráfico malicioso en internet, a través de varias pruebas y simulaciones. Los resultados demuestran que el árbol de clasificación creado es fiable, porque arrojó altos niveles de efectividad, dado que se guía por parámetros inevitables por un atacante y que, por lo tanto, siempre sería detectado.

Otro estudio analizado, en relación con la técnica de árbol de decisión con el algoritmo J48, es aquel efectuado por Firdausi et al. (2010), cuya finalidad es el desarrollo de una prueba de concepto para la detección automatizada de malware, basada en el comportamiento y en la aplicación de cinco técnicas clasificadores de aprendizaje automático, a saber: k-Vecinos más cercanos (kNN), Naïve Bayes, J48 Decision Tree, Support Vector Machine (SVM) y Multilayer

Perceptron Neural Network (MLP). La preparación de los datos se efectúa con dos conjuntos de datos que tienen el formato de archivos binarios de Windows Portable Executable (PE) y que distingue entre datos de malware (de Indonesia) y benignos; con los cuales se realiza un análisis dinámico de los datos para monitorear su comportamiento, usando la herramienta en línea Anubis. Se prosigue con el preprocesamiento de datos para la creación de archivos de formato de archivo de relación de atributos (ARFF), dado que las pruebas y experimentos se efectúan utilizando el software Weka y que permiten establecer que el árbol de decisión J48 logra el mejor rendimiento en función de su recuperación (tasa de verdaderos positivos), tasa de falsos positivos, precisión (valor predictivo) y exactitud.

6.1.5. Deep Neural Network (DNN)

Otra de las técnicas de Machine Learning identificadas con el desarrollo de este proyecto, es la Red Neuronal Profunda (DNN por su sigla en inglés). Una red neuronal profunda es una red neuronal artificial (ANN) con varias capas ocultas entre las capas de entrada y salida; al igual que en las ANN poco profundas, los DNN pueden modelar relaciones no lineales complejas.

El propósito principal de una red neuronal es recibir un conjunto de entradas, realizar cálculos progresivamente complejos en ellas y dar salida para resolver problemas del mundo real como la clasificación. En el aprendizaje profundo, el número de capas ocultas, en su mayoría no lineales, puede ser grande, en el orden de los cientos o miles de capas, lo que hace que su entrenamiento tenga un alto costo computacional.

Este tipo de red fue analizada por Lu et al. (2019), quienes realizaron esta investigación con el objetivo de observar cómo se comporta el malware para luego clasificarlos a través de algoritmos tradicionales de la máquina y los algoritmos de red neuronal profunda, para luego

comparar el resultado con el sistema de clasificación de malware basado en reglas convencionales, en dos tipos diferentes de conjunto de tareas. Después de hacer la comparación, se encontró que el aprendizaje profundo se desempeña mejor en cuanto a precisión y versatilidad. Es más seguro analizar los datos mediante un campo de entrenamiento que hacer un campo de búsqueda profunda. Para comprobar lo anterior se construyó un modelo de GAN basado en un texto y en una imagen de datos con software malicioso y diseñar una especie de nueva arquitectura de predicción de malware que muestre cierta viabilidad en las bases de datos probadas. Mediante las herramientas MalDeepNet (TB-Malnet y IB-Malnet) para las tareas de clasificación del comportamiento dinámico del malware, construyendo un algoritmo de clustering de la familia basado en el aprendizaje profundo con técnicas diferentes de ofuscación y transformación de código con gran precisión.

Ahora bien, el objetivo general con el trabajo de Zufiaurre (2019) es desarrollar tres sistemas mediante el entrenamiento de modelos clasificadores Shallow Learning (aprendizaje tradicional) y una red neuronal entrenada usando Deep Learning (aprendizaje profundo); enfocados en la detección y clasificación de aplicaciones malware o benignware para el sistema operativo Android. La creación de los tres sistemas basados en el aprendizaje tradicional consistió en la combinación de siete modelos de clasificadores y cinco técnicas de seleccionadoras de características; los cuales, a su vez, fueron seleccionados según las tres métricas de evaluación determinadas, a saber: el modelo Random Forest Classifier fue el de mayor exactitud y el modelo Multi-layer Perceptron Classifier revela los resultados más altos en precisión. Por otra parte, para la compilación de la red neuronal se elige la función binary cross entropy, Adam como función de

optimización, el entrenamiento se efectúa con las funciones de la librería Keras[18] y la calidad se evalúa con la exactitud de la red, la precisión, la sensibilidad y, por último, con la matriz de confusión. En definitiva, los sistemas Shallow Learning, según las métricas evaluadas, superaron a la red neuronal profunda que generó un entrenamiento y una predicción más inexacta y costosa.

Sumado a lo anterior, se trae a colación el estudio de González y Vázquez (2015), quienes se propusieron encontrar un método de clasificación de malware que detectara la mayor cantidad posible de muestras maliciosas, basado en el número de veces que el programa (APIs) llama a diferentes funciones de cada biblioteca de enlace dinámico (Dynamic Link Libraries, DLL) y que sean de una misma familia o derivadas de ellas. Éstas tienen estructuras similares y por lo tanto comparten una gran cantidad de bibliotecas y funciones del sistema. Todas estas características de los programas maliciosos fueron ordenadas de forma numérica para ser llevadas a unos clasificadores en específico como las redes neuronales artificiales. Este enfoque se eligió debido a que los métodos de clasificación seleccionados para los experimentos están basados en reconocimiento de patrones, por lo que trabajan exclusivamente con valores numéricos, y durante el análisis de la información que es posible extraer de un ejecutable, surgió la idea de contar la cantidad de funciones que los programas llaman de cada biblioteca y con esos valores crear un vector, que podría ser utilizado como entrada para distintos clasificadores y tipos de redes neuronales. Las muestras utilizadas en esta investigación sólo son de troyanos y gusanos y fueron obtenidas paginas Offensive Computing y VX Heavens, utilizan un ambiente virtual con un sandbox, utilizan desensamblador IDA Pro, detectores de archivos comprimidos como PEid y RDG Packer Detector, también utilizan analizadores como OllyDgb como apoyo al IDA Pro y

[18] Keras es una biblioteca de software de código abierto que proporciona una interfaz Python para redes neuronales artificiales.

para identificar las funciones importantes se empleó UPX. PECompact y ASPack y el software ImpRec. El desempeño se hace con 245 muestras de 24 familias en una comparación entre la técnica Distancia Euclidiana y una red perceptrones multicapa, con una y dos capas ocultas, usando los algoritmos de aprendizaje de Retropropagación y LevenbergMarquardt. Obteniendo el mejor resultado es de 89.25% y se obtuvo con el perceptrón multicapa, con una arquitectura de una capa oculta, entrenando con el algoritmo Levenberg-Marquardt y empleando el 80% de las muestras para el entrenamiento.

El trabajo de Chamorro (2020), enfocado en comprobar la potencia de dos técnicas de Machine Learning y con una de estas, crear un modelo más robusto y efectivo contra ficheros malware que favorezca la ciberseguridad, sin la interacción humana. Para la detección del malware se utiliza el método basado en heurísticas y una técnica de análisis estática con un conjunto de datos propio, conformado con un dataset de ficheros Portable Executable (PE) infectados de Windows, del cual se extrajeron los ficheros benignos con una máquina virtual de Windows 10 y se realiza la división de este conjunto en dos grupos, uno para la fase de entrenamiento y otro, para el test. Los entrenamientos de la fase I, se realizan con una instancia configurada, a través de Google Cloud, utilizando los modelos Light Gradient Boosting Machine (LGBM) y Deep Neuronal Network (DNN), los cuales son evaluados con las métricas de exactitud, clasificación errónea (CE), sensibilidad y precisión; a partir de las métricas base de la matriz de confusión. Por motivos de la pandemia Covid-19, Google reduce la cuota límite de Google Cloud a 8 CPUs, por lo cual se redefine y disminuye el conjunto de datos con el uso de librerías pandas y se efectúan las modificaciones para usar una instancia con 12 CPUs, antes de 24 CPUs. Así las cosas, se opta por mejorar el DNN porque arrojó valores bajos en las métricas de exactitud, precisión y sensibilidad y un valor mayor en cuanto a CE. Una de las alternativas de robustecimiento y

efectividad de la técnica DNN, es aplicar la técnica Ensemble Methods, con la combinación de dos grupos, en este caso, el modelo Random Forest (weak learner) y el DNN, como strong learner. Con este nuevo modelo ensamblado, se obtiene tasas más altas de detección positiva y menores tasas de errores.

Por otra parte, se tienen la investigación de Asad et al. (2020), cuyo enfoque con la misma es determinar el modelo con más precisión para predecir si las maquinas individuales serían infectadas por malware; mediante el uso de algoritmos de aprendizaje automáticos supervisados y de aumento de gradiente. El conjunto de datos está conformado por los datos de predicción de malware de Microsoft, divididos en dos subconjuntos, uno para el entrenamiento y el otro, para las pruebas; teniendo en cuenta que con el primero, se realiza una matriz de calor de correlación. Para la predicción del modelo se usan tres algoritmos diferentes: 1) LightGBM, un algoritmo que impulsa el gradiente y que mejora sus resultados usando la técnica de validación cruzada de K-Fold; 2) Decision Tree Classifier, un algoritmo clasificador de árbol de decisión, para el cual se reprocesa el conjunto de datos de entrenamiento y 3) Neural Network, usado para construir y entrenar el modelo. Estos algoritmos son evaluados según su rendimiento con la métrica puntuación del área bajo la curva ROC (AUC), teniendo en cuenta que valores inferiores a 0,5 se consideran malas y las puntuaciones entre 0,5 y 0,7 se consideran medias. Las puntuaciones de 0,7 y superiores se consideran buenas. Desde esta perspectiva, se determina que LightGBM, tiene la mayor precisión, seguido de la red neuronal que arroja un valor AUC satisfactorio, pero que no es eficiente porque tiene un mayor consumo de memoria y tiempo de ejecución; y el clasificador de árbol de decisión obtiene un rendimiento medio y el tiempo de ejecución más alto.

Ahora bien, hasta aquí se han explicado los aspectos de mayor relevancia encontrados en los antecedentes investigativos identificados por su uso de métodos de ML de mayor aplicación

encontrados en la revisión de literatura, los cuales reportan la aplicación y ejecución de experimentos mediados con la utilización de técnicas basadas en Machine Learning para la detección y clasificación de todo tipo de malware. Por este motivo, en las siguientes líneas de exponen aquellos antecedentes enfocados específicamente en malware ransomware.

6.1.6. Técnicas de Machine Learning en la literatura aplicadas específicamente en la detección del Malware Ransomware

El malware ransomware tiene características específicas y usa vectores de infección determinados que lo diferencian de otro tipo de malware; por esto se consideró importante identificar los aspectos más relevantes de las técnicas de ML adaptadas para la detección de este malware particularmente.

Para probar la efectividad de su propuesta, Maimó (2019), realizara una serie de experimentos utilizando algunos de los malware más peligrosos y recientes: WannaCry, Petya, BadRabbit y PowerGhost. El primer paso fue recrear, usando OpenICE y máquinas virtuales, un entorno real clínico. Con respecto a los resultados en clasificación con este conjunto de datos, el método semisupervisado que mejor rendimiento obtuvo en los experimentos fue OC-SVM (máquina de vector soporte monoclase), entrenada con tráfico normal, que obtuvo una precisión de 0,9232, una sensibilidad de 0,9997 y una tasa de falsos positivos de 0,046 a la hora de clasificar tráfico conjunto (normal y anómalo). En lo que respecta al método supervisado, cada uno de los métodos obtenía un rendimiento aceptable para cierta combinación de hiperparametros. De entre dichos parámetros cabe destacar por su importancia el tamaño de la ventana de tiempo usada para la generación del vector de características, que terminó siendo fijada en 10 segundos. Se seleccionó Naive Bayes por ser el más sencillo de los tres evaluados y ofrecer un rendimiento similar para

una ventana de 10 segundos, teniendo especialmente buen comportamiento en detectar ransomware desconocido. Los resultados muestran una precisión y sensibilidad del 99,99%. Es importante recordar que en las reglas que activan la mitigación se tienen en cuenta ambos predictores (OC-SVM y Naive Bayes) para tomar la decisión. Para ello se decidió utilizar virtualización de funciones de red (NFV) que proporciona flexibilidad y dinamismo en la infraestructura al separar la capa hardware del software, y redes definidas por software (SDN) que nos aporta control de las comunicaciones en tiempo real y bajo demanda.

En esta misma línea argumentativa, Benavides y Roa (2018) proponen una herramienta de extracción automática de información de malware, para determinar si la captura de pantalla del ataque del ordenador víctima, la ventana emergente recortada y el reconocimiento de patrones corresponden a una de las muestras de ransomware almacenadas previamente y que están involucradas en un ataque informático; a partir de la aplicación del Reconocimiento Óptico de Caracteres (OCR), el volcado de la memoria RAM y la extracción de archivos relevantes. En este caso, se explican teóricamente los conceptos de malware y ransomware y además, se utilizan las siguientes técnicas para crear la herramienta de extracción: la técnica de volcado de memoria con Dumplt, el algoritmo SIFT para el procesamiento de imágenes y Pytesseract como técnica de OCR. La herramienta diseñada e implementada, sigue este procedimiento: obtención de la captura de pantalla y su respectiva región de interés, búsqueda de patrones característicos de ransomware, tratamiento y transformación morfológica de la imagen, se realiza el OCR, búsqueda de la información relacionada con el ataque en los ficheros almacenados en el dispositivo víctima del ataque, ejecución del volcado de la memoria RAM, almacenamiento de la información en una memoria USB comprimida en un archivo.zip y se finaliza con el análisis de la memoria volátil con el programa Volatility. La experimentación de esta técnica, se realiza con un computador de

escritorio preparado como víctima, porque es infectado con un total de 13 muestras de ransomware diferentes; hallándose con el respectivo análisis de escalabilidad que las muestras del ransomware CryptoLocker, TeslaCrypt, Cerber y Hermes arrojaron resultados positivos en todas sus funcionalidades; en contraste con las muestras Crysis, Wannacry y GandCrab que no fue posible ejecutar la aplicación, ya que la memoria USB fue cifrada. Para determinar si la captura de pantalla del ataque del ordenador víctima, la ventana emergente recortada y el reconocimiento de patrones corresponde a una de las muestras de ransomware almacenadas previamente y que están involucradas en un ataque informático. Hallándose con el respectivo análisis de escalabilidad que las muestras del ransomware CryptoLocker, TeslaCrypt, Cerber y Hermes arrojaron resultados positivos en todas sus funcionalidades; en contraste con las muestras Crysis, Wannacry y GandCrab que no fue posible ejecutar la aplicación, ya que la memoria USB fue cifrada.

Un tercer estudio encontrado con respecto a esta temática, es la investigación de Bazante (2019), orientada en proponer un modelo de aprendizaje automático para la filtración de algunos atributos de los ransomware Cryptolockery y Wannacry; a partir del análisis dinámico del comportamiento de estos malware con el sistema automatizado Cuckoo Sandbox, en un ambiente de implementación controlado (Sandboxing) que permita la detección de estos ataques, por medio de una clasificación automática de los mismos, identificando las acciones realizadas por el ransomware en el sistema infectado. Las muestras se obtienen de un repositorio alojado en GITHUB, las cuales son preparadas según los atributos de esta dataset, se procede con la colocación de las muestras en la interfaz Web de Cuckoo Sandbox, se efectúa el respectivo análisis dinámico y se presentan los artefactos extraídos como scripts, mediante los reportes generados con esta herramienta. Se continúa con la creación de un modelo de Machine Learning, usando dos sistemas, uno con Windows 7 y otro la versión 10 y la herramienta de minería de datos Rapidminer;

obteniendo un modelo predictivo basado en la técnica de árboles de decisión. Después, se aplican diversas herramientas de filtrado, por ejemplo, Filter Examples, y posteriormente, se consolida el modelo para predecir qué tipo ransomware infecta el sistema, usando el árbol de decisión para el entrenamiento y los operadores Apply Model y Perfomance para el test, apoyado además, en un algoritmo de validación cruzada. La evaluación del modelo de Machine Learning se realiza con las métricas de tiempo de ejecución del análisis (duration), la calificación dada por la herramienta basado en el comportamiento y firmas (score) y la precisión, teniendo en cuenta que este último arrojó un valor igual 97,50%; por lo tanto, se evidencia que este modelo, logra predecir el tipo de ransomware que ataca un determinado sistema.

Por otra parte, Herrera et al. (2019), propusieron la creación de un Dataset público e indica los parámetros que han sido seleccionados desde un análisis estadístico de las características seleccionadas como el comportamiento de red y uso de procesos del sistema víctima durante la fase de infección de un sistema Windows XP y Windows 7. La tendencia actual del ransomware es presentarse como un servicio bajo demanda en el que se incluyen paquetes con exploits kits para vulnerar sistemas y perpetrar ataques dirigidos. En el contexto de este desarrollo, se identificó una serie de parámetros de detección y prevención, mediante una diversidad de herramientas de análisis del ransomware como Anubis, VirusShare, Virus Total, Proccess Monitor, Watchdog Module, pero se basan principalmente en Cuckoo Sanbox. También desde el autor Mohammad en 2019, realiza un análisis de un conjunto de parámetros relacionados con ataques ransomware, siendo las métricas más utilizadas: región de convergencia (ROC) contra el cifrado de archivos, utilización de CPU, tasa de positivos verdaderos (TPR), tasa de falsos positivos (FPR), precisión y recuperación. Por otro lado, según el sistema RWGuard (Mehnaz, 2018), los parámetros que

pueden influir en la detección de ransomware son: los paquetes requeridos de entrada y salida, el comportamiento y procesamiento de CPU.

Por otro lado, dentro de las investigaciones a nivel de detección, los principales parámetros que se toman en cuenta son claves de registro, actividades de entrada/salida de archivos de sistema, actividad de procesos, entropía, llamadas a funciones API (Chen, 2017), actividad de red, características de red (protocolo, direcciones IP fuente y destino, puertos, paquetes, duración). Se implementa un ambiente de pruebas controlado y un análisis de muestras de ransomware a través del filtrado de sus atributos, de tal manera que con la información recolectada se construye el dataset. El Dataset consta de 10 columnas, las cuales representan los identificadores de objetos y características tomadas en cuenta en el proceso de análisis, extracción de información y filtrado de la misma, tomados del archivo report.json dado por Cuckoo Sandbox. El archivo ".json" consta de 15 objetos y en cada objeto existen características que a su vez contienen otras anidaciones. Los objetos analizados fueron tres con sus características: procmemory (file, urls, pid), network (hosts, dns) y behavior (Processes, summary), y cada una de ésas características contiene subcaracterísticas. El Dataset propuesto considera parámetros relacionados al comportamiento de registros durante un ataque, los procesos de memoria y de red que permiten identificar si el ataque se relaciona con una muestra de CryptoLocker, CryptoWall, PetrWrap, Petya o WannaCry (Herrera et al., 2019).

Finalmente, el estudio de Vivanco-Toala et al. (2020), cuyo objetivo es realizar un estudio exploratorio de las estrategias para proteger las redes empresariales de las infecciones Ransomware, para lo cual, se plantea la necesidad de establecer un esquema de que tipos de infecciones ransomware son las más comunes, a través de una matriz de diferenciación de los tipos de ataques (TeslaCrypt (Dirigido a los videojuegos, cifrándolos) se distribuye a través de los kits

de explotación Angler, Sweet Orange y Nuclear, además no utiliza algoritmos asimétricos RSA-2048 sino AES, CTB-Locker (Utiliza la red TOR para su infraestructura, exclusivamente para servidores C2) se distribuye a través de descargas, correos. Se extinguió también a través de características, tales como, un servidor de control basado en TOR y direcciones Bitcoin generadas automáticamente, únicas para cada víctima, donde, el rescate sigue siendo vendido a criminales cibernéticos, CryptoWall (Infecta por correo electrónico Spam con malware en el archivo adjunto y otra a través de los sitios web infectados con Angler Exploit Kit) fue la primera variante de ransomware desde el año 2014, que sólo aceptó pagos de rescate en Bitcoin y empleó clave pública RSA de 2048 bits para el cifrado de archivos importantes, CryptoLocker (cambia el nombre de todos los archivos, carpetas y los cifra), entra en las empresas por medio del correo electrónico, es decir, si el usuario hace clic en el ejecutable adjunto al correo inmediatamente comienza a analizar las unidades de red, utilizó cifrado RSA de 2048 bits y por último Locky del tipo Crypto Ransomware (se propaga a través de correos electrónicos de Spam que incluye documentos maliciosos de Microsoft Office o archivos adjuntos comprimidos, por ejemplo .rar .zip). Éstos archivos adjuntos maliciosos contienen macros o archivo JavaScript para la descarga del malware Locky y puede llegar a cifrar más de 160 tipos de archivos diferentes, este ransomware se ha distribuido utilizando el Kit de Explotación Nuclear). Con las características del ataque, el método del rescate y las estrategias recomendadas se plantean pasos para implementar la seguridad en las PYMES. El detallar los tipos de ataques ransomware descritos en el objetivo planteado, permiten a las grandes y pequeñas empresas PYMES, conocer las características y comportamientos de los ransomware, es decir la forma de propagación del ataque, ya sea a través de los correos electrónicos como spam con archivos adjuntos, por publicidad infectada o por sitios web infectados con angler

exploit kit, para luego, utilizar en el cifrado de información de las víctimas, algoritmos simétrico AES o asimétrico RSA, dependiendo del tipo de ransomware empleado.

6.2. Caracterización del Malware Ransomware y las principales técnicas que existen para su detección

En este apartado se analizan las principales características y patrones del malware ransomware, según la información obtenida de la literatura. También se caracterizan las técnicas del ML utilizadas para su detección teniendo en cuenta los resultados reportados en los casos de aplicación en función de las métricas de eficacia.

Es preciso clarificar que, para la construcción de la caracterización de las técnicas, se tuvieron en cuenta cuatro estudios referenciados en la sección anterior y que cumplieron con los siguientes criterios de inclusión: 1) La aplicación de técnicas de aprendizaje automático de detección y clasificación, 2) El análisis explícito y específico de muestras del malware ransomware y 3) Los resultados obtenidos reportan las métricas de mejor rendimiento. Por este motivo, se descartó para este análisis, la publicación de Vivanco-Toala et al. (2020), porque si bien describe y caracteriza una serie de familias del malware ransomware, no reporta experimentos con resultados cuantitativos en relación a las técnicas de Machine Learning para comprobar su efectividad.

Tabla 2.

Características intrínsecas propias del malware ransomware.

Nombre	Extensión	Nombres de Archivos de notas de Ransomware	Algoritmo de Encriptación	Técnicas de Acceso	Método de rescate
CryptoLocker	.encrypted .ENC	Inicia cada vez que se enciende una computadora infectada, también le proporcionará una ventana de pago que disminuye rápidamente.	AES-256 simétrico y el RSA-2048 asimétrico	Correos de spear phishing con documento adjunto malicioso, puertos RDP[19]	El sistema de pagos es por Bitcoins y se encuentra alrededor de $300[20]
Petya	.encrypted	YOUR_FILES_ARE_ENCRYPTED.TXT	AES-256	Sobrescribe el MBR (Master Boot Record) del PC, Correos de spear phishing con documento adjunto malicioso con un enlace de descarga de Dropbox que aparentemente redirige al currículum de alguien[21].	Cobro a las víctimas de 300 USD en bitcoines a cambio de liberar los archivos. A quien no pagaba a tiempo se le doblaba el precio de la clave de descifrado[22]
TeslaCrypt	.vvv .ecc .exx .ezz .abc .aaa .zzz .xyz .micro .xxx .ttt .mp3	HELP_TO_SAVE_FILES.txt Howto_RESTORE_FILES.html RECOVER<5_chars>.html RECOVER<5_chars>.png RECOVER<5_chars>.txt _how_recover+<random 3 chars>.txt or .html, help_recover_instructions+<random 3 chars>.BMP or .html or .txt, _H_e_l_p_RECOVER_INSTRUCTIONS+<random 3 char>.txt, .html or .png Recovery+<5 random chars>.txt, .html, e.g., Recovery+gwote.txt RESTORE_FILES_<random 5 chars>.TXT , e.g. restore_files_kksli.bmp, HELP_RESTORE_FILES_<random 5 chars>.TXT , e.g. help_restore_files_kksli.bmp, HOWTO_RECOVER_FILES_<random 5 chars>.TXT. e.g. howto_recover_files_xeyye.txt, HELP_TO_SAVE_FILES.txt or .bmp	AES (256) + ECHD + SHA1	Fallo de Adobe Flash, Ataques a servicios RDP, Correos de spear phishing con documento adjunto malicioso y Explotación de vulnerabilidades en soluciones VPN[23]	El rescate de pago se puede realizar por Sistema Paypal o por Bitcoin, y cuesta alrededor de $1000 el ransoware por PayPal y $500 por Bitcoin[24].

Nota: adaptado, según la literatura, principalmente de Bazante (2019).

[19] Adaptado AO Kaspersky Lab (2021).
[20] Tomado de Vivanco-Toala et al. (2020).
[21] Adaptado de Sumalapao (2016).
[22] Tomado de Latto (2020).
[23] Adaptado AO Kaspersky Lab (2021).
[24] Tomado de Vivanco-Toala et al. (2020).

Tabla 2.

Características intrínsecas propias del malware ransomware (continuación)

Nombre	Extensión	Nombres de Archivos de notas de Ransomware	Algoritmo de Encriptación	Técnicas de Acceso	Método de rescate
WannaCry	.wcry .wncry .WNCRY .WCRY	@Please_Read_Me@.txt	AES-256 y RSA-2048	Explotación de aplicaciones web expuestas en internet como Tomcap o weblogic, Ataques a servicios RDP, Correos de spear phishing con documento adjunto malicioso y Explotación de vulnerabilidades en soluciones VPN[25]	Cobro a las víctimas de 300 USD en bitcoines a cambio de liberar los archivos. A quien no pagaba a tiempo se le doblaba el precio de la clave de descifrado[26].
BadRabbit	.encrypted	infpub.dat, cscc.dat y dispci.exe, Readme.txt	AES en modo CBC	Sobrescribe el MBR (Master Boot Record) del PC, Correos de spear phishing con documento adjunto malicioso, con una actualización de Flash, basado exclusivamente en la ingeniería social[27].	Cobro a las víctimas de 300 USD en bitcoines a cambio de liberar los archivos. A quien no pagaba a tiempo se le doblaba el precio de la clave de descifrado[28]

Nota: adaptado, según la literatura, principalmente de Bazante (2019).

En este sentido, se evidencia según la información de la tabla 2, que las principales versiones del malware ransomware halladas en la literatura y que han sido analizadas por los autores, son: Wannacry, CrytoLocker, Petya, TeslaCrypt y Badrabbit. Con frecuencia para valorar el potencial y la precisión de las técnicas y modelos de detección y clasificación basados en Machine Learning, los investigadores efectúan análisis de software malicioso, mediante la combinación de los tipos de ransomware mencionados, con otros más, tales como: Cerber, Hermes, Crysis, GandCrab, JigSaw, Vipassana, BTCware, Saturn, Locky, Jaff, CryptoWall.

Antes de continuar con la caracterización intrínseca de los malware ransomware encontrados en la literatura, es pertinente indicar que la tabla 2, se concentra en la descripción de

[25] Adaptado AO Kaspersky Lab (2021).
[26] Tomado de Latto (2020).
[27] Adaptado AO Kaspersky Lab (2021).
[28] Tomado de Latto (2020).

los softwares maliciosos Wannacry, CrytoLocker, Petya, TeslaCrypt y Badrabbit; sin embargo, la información plasmada en la tabla 2, se deriva de otra tabla de mayor extensión y más completa que puede ser revisada en el anexo A de la presente investigación.

Según las características intrínsecas de los malware ransomware estudiadas se revela que, en materia de extensiones, algunos softwares maliciosos tienen extensiones asociadas con la encriptación y otros con los archivos que cifran al momento de la infección. Al respecto, es posible precisar que en el caso de las extensiones, los ransomware CryptoLocker, Petya y Badrabbit comparten el mismo tipo de encriptación con la extensión *.*encrypted*, el cual se manifiesta durante el cifrado del sistema, cuando a todos los nombres de los archivos encriptados se les agrega dicha extensión, lo que dificulta la identificación de los archivos afectados. Aunque también se añade un archivo de texto en el escritorio, para informar a la víctima que sus archivos han sido encriptados y que podrá recuperarlos una vez, efectúe el pago de la recompensa solicitada.

Por otra parte, se visualiza conforme con los datos de la tabla 2, que los malware ransomware WannaCry y TeslaCrypt tienen en común, que estos han sido usados, por medio de diversas extensiones; las cuales funcionan de forma similar que con la extensión *encrypted*, lo que significa que añade a los nombres de los archivos encriptados sus respectivas extensiones, principalmente la extensión *.ecc en el caso del ransomware TeslaCrypt y wncry en lo concerniente con WannaCry. Así mismo, emiten un archivo de nota, con el cual se le notifica a la víctima la encriptación de sus archivos y las instrucciones a seguir para su eventual acceso y recuperación.

Sumado a lo anterior, los malware ransomware CryptoLocker, Petya, WannaCry y TeslaCrypt emplean el mismo estándar de cifrado avanzado, es decir el estandar AES-256; lo que significa que utiliza el cifrador AES-256 y el proceso de encriptado de bloques de datos se produce

en 256 bits y es más fuerte, por lo tanto, en este caso es usado para corromper los archivos del sistema víctima de una forma más rápida y así proceder a agregar la respectiva extensión de archivo, según el virus infectado. Adicionalmente, los ransomware Badrabbit, WannaCry y CryptoLocker utilizan el algoritmo de encriptación RSA-2048; en general, con cifrado fuerte los archivos no pueden ser desencriptados sin la clave única usada por los cibercriminales con dicha finalidad.

De acuerdo con las técnicas de acceso, se evidencia conforme con los datos de la tabla 2, que todos los virus ransomware caracterizados en la misma, comparten el método asociado con el envío de correos de *spear phising* con documento adjunto malicioso, lo que significa que se produce una estafa a través de un correo electrónico, con el propósito de tener un acceso no autorizado a los datos confidenciales de la víctima. Por otra parte, los malware ransomware CryptoLocker, TeslaCrypt y WannaCry infectan los sistemas de las víctimas por medio de ataques a los servicios RDP, es decir a los Servicios de Escritorio Remoto (del inglés Remote Desktop Services), lo cual se realiza con frecuencia para explotar y acceder sin autorización a las redes empresariales, así como para amplificar el ataque efectuado. En forma similar, la explotación de vulnerabilidades en soluciones de Red Privada Virtual (VPN por su traducción en inglés), se ejecuta con los virus TeslaCrypt y Wannacry, ocasionando el despliegue de ataques internos. Adicionalmente, otras técnicas de acceso están relacionadas con las fallas en Adobe Flash, enlaces de descarga aparentemente confiable porque puede llevar el nombre de una aplicación reconocida, la sobre escritura de la Master Boot Record (MBR) del PC y la actualización de la aplicación Flash por medio de una metodología de ingeniería social. Al respecto de esta última, según Cusaria y Fagua (2017), la ingeniería social se define como “el acto de manipular a una persona para que lleve a cabo una acción que puede ser o no lo más conveniente para el “objetivo”; [para] obtener

información o conseguir algún tipo de acceso" (p. 40). Asociado con lo anterior, en cuanto al método de rescate, de forma unánime, todos los malware ransomware analizados utilizan la misma metodología para que las víctimas de la infección recuperen los archivos encriptados, es decir la solicitud de un pago en bitcoins, esto con la finalidad de evitar la identificación y el rastreo de los cibercriminales. Dicho pago oscila entre 300 y 1.000 dólares, un monto que deben pagar las victima en un tiempo determinado, porque en caso contrario, pierden la información cifrada con el ransomware.

En términos generales, los malware ransomware estudiados en la literatura analizada para efectos de la presente investigación, permiten establecer que sus características intrínsecas revelan principalmente, similitudes en relación con las extensiones, proceso de encriptación, la modalidad de notificación efectuada con la víctima, y con respecto a la rapidez y la optimización que se le ha dado a los algoritmos criptográficos, ya que los actuales cifradores aceptados por la comunidad científica, por sus avances y alto nivel de seguridad están siendo utilizados como un ransomware, debido a sus propia fuerza de encriptación.

En línea con la construcción de este estudio, a continuación en la tabla 2, se detallan y analizan las principales técnicas y métodos de Machine Learning, que de acuerdo con la literatura revisada, han sido utilizadas para la detección, clasificación y análisis del malware ransomware; siendo la Máquina Vector Soporte Monoclase (OC-SVM), Naive Bayes, los árboles de decisión, el volcado de memoria RAM, y el Reconocimiento de Óptico de Caracteres (OCR) las técnicas más empleadas, junto con los paradigmas de Virtualización de Funciones de Red (NFV) y las Redes Definidas por Software (SDN) para el desarrollo y pruebas seguras en relación con la mitigación de la propagación del ransomware, al aislar los equipos y redes de prueba, facilitando reemplazar los dispositivos infectados después de la validación.

En esta instancia y antes de proseguir con la caracterización de las técnicas de aprendizaje automático, es importante indicar que para mejorar la comprensión de esta temática, la organización de la información de la tabla 3 se efectuó con relación a las 4 diferentes técnicas de detección propiamente reportadas en los trabajos seleccionado por los criterios de inclusión antes referenciados; la tabla presenta los tipos malware ransomware detectados por el sistema, los atributos que se utilizan como parámetros de entrada, describe cada uno de los modelos de Machine Learning utilizado en las diferentes etapas de clasificación y detección, las diversas fases del proceso y los resultados reportados por los autores.

Tabla 3.

Caracterización de las técnicas para la detección del malware ransomware

Malware ransomware analizados	Características utilizadas para la detección del malware ransomware	Técnicas de detección y clasificación evaluadas	Proceso de deteccion y clasificación	Resultados de las métricas
WannaCry **Petya** **BadRabbit**	De acuerdo con los aportes de Maimó (2019), los ransomware se detectaron mediante la utilizacion de caracteristicas de contexto (calculadas a partir de la ventana de flujos) y características locales (calculadas a partir del último flujo). Dichas características, son: • % de caudales. • Media y stddev de las duraciones de flujo. • Media y stddev de tiempo entre dos flujos consecutivos. • Número de direcciones IP de destinos diferentes. • Entropía de las direcciones IP de destino. • Suma, máx., Mín., Media, desv. Estándar y mediana del total de paquetes. • Suma, max, min, mean, stddev y mediana de los paquetes fuente. • Suma, max, min, media, stddev y mediana del total de bytes. • Suma, máx., Mín., Media, desv. Estándar y mediana de los bytes de origen. • Suma, max, min, media, stddev y mediana de la carga total. • Suma, máx., Mín., Media, desv. Estándar y mediana de la carga de la fuente. • % de puertos de origen / destino> 1024. • % de puertos de origen / destino <1025. • Número de puertos de origen y destino diferentes. • Número de direcciones IP de destino, diferentes. • Entropía de puertos de origen y destino. • Entropía de la IP de destino. • Mediana de la duración. • Protocolo utilizado (TCP, UDP, ARP). • Estado (INT, RST, FIN, CON).	Para la detección de malware ransomware, Maimó (2019), evaluó las siguientes tecnicas de detección de anomalías, • Máquina de vectores de soporte de una clase (OC-SVM) • Factor de valor atípico local (LOF) • Bosque de aislamiento (IF). Así mismo valoró los siguientes algoritmos de clasificación: • Red neuronal (NN) • Naive Bayes (NB) • Random Forest (RF). Además, evaluó la mitigación de la propagación del ransomware al aislar y reemplazar los dispositivos infectados mediante una combinación entre los paradigmas Virtualización de Funciones de Red (NFV) y las Redes Definidas por Software (SDN). Finalmente, Maimó (2019), identificó que la mejor opción, es una combinación entre la Máquina de Vector Soporte Monoclase (OC-SVM) de tipo semisupervisado para la detección de anomalías y Naive Bayes de tipo supervisado para la clasificación de ransomware conocido y no visto.	• Se diseñó y desplegó un ambiente controlado con Entornos Clínicos Integrados (ICE), para obtener un conjunto de datos adecuado que permitirá realizar los experimentos de detección. • Se realizó la captura de seis horas de tráfico de red limpio en el escenario diseñado al respecto. Y conjuntos de datos capturados durante la propagación, con el tráfico del ransomware. • Se fijó una ventana de 10 segundos, para calcular caracteristicas agregadas, según los flujos recibidos durante los últimos 10 segundos. • Definición de hiperparámetros para el proceso de selección de la técnica de detección y el clasificador de ransomware. • Se evaluaron las técnicas y algoritmos para la selección de la combinación para la detección y clasificación de malware ransomware en el escenario creado. • Creación de un nuevo ambiente controlado, después de la selección de los métodos de detección y clasificación, para implementar la interfaz de equipos ICE sobre diferentes máquinas virtuales. • Medición del tiempo de implementación del nuevo escenario, de infección y mitigación.	**Resultados de NFV y SDN** • Precisión y sensibilidad: 99,99%. **Resultados de OC-SMV** • F1: 95,96% • Precisión: 92,32%. • Recuperación: 99,97% en la detección de anomalías. • Relación de falsos positivos (FPR): 4,6%. **Resultados Naive Bayes (NB)** • Precisión de clasificación: oscila entre 98,97% y 100%.

Nota: adaptado de Maimó (2019).

Tabla 3.

Caracterización para detección del malware ransomware (continuación)

Nombre del malware ransomware	Características utilizadas para la detección del malware ransomware	Técnicas de detección y clasificación evaluadas	Proceso de detección y clasificación	Resultados de las métricas
WannaCry **TeslaCrypt** **CryptoLocker**	De acuerdo con Benavides y Roa (2018), la evaluación de las muestras de ransomware, se efectuó con las siguientes características clave: • Correo electrónico. • URL • Dirección de Bitcoin • Dinero del rescate. • Fondo de pantalla. • Nota de rescate • Ventana emergente con las características de ransomware conocidos. • Archivos con extensiones .bmp y .html. • Cifrado de la unidad de almacenamiento.	• Reconocimiento Óptico de Caracteres, OCR. • Volcado de memoria RAM con Dumpit y con Volatility.	• Creación e infección de un ambiente controlado, es decir, un Sistema operativo Windows 7 de 64bits, 4GB de memoria RAM, configuraciones y aplicaciones comunes en usuarios caseros (Microsoft Office, Adobe Reader), para simular un entorno real de la víctima y se generaron ficheros con distintos formatos (texto, imágenes y videos). • Uso de una Red de Ciberseguridad de la FIS-EPN, con firewall y reglas de control de acceso y control de servicios, para realizar los experimentos. • Reconocimiento óptico de caracteres, OCR de la ventana emergente, reconocimiento de patrones, reconocimiento de bordes que detecta los cambios en la intensidad de luz en la imagen (captura), recorte en la imagen original. • Extracción de descriptores SIFT para encontrar 35 puntos de coincidencias de cada patrón ransomware en la imagen extraída. • Preparación de la imagen resultante mediante una transformación morfológica y cambios de color a escala de grises, para facilitar y mejorar el reconocimiento óptico de caracteres, • Ejecución del Reconocimiento Óptico de Caracteres (OCR) del fichero de texto con las instrucciones para recuperar la información secuestrada. • Se realiza una búsqueda de ficheros con extensiones .txt y .html sobre una máquina infectada, para dejar solo aquellos con las instrucciones proporcionadas por el cibercriminal y se procede con su almacenamiento en una memoria USB. • Se realiza el volcado de la memoria RAM del dispositivo infectado con la herramienta Volatility, para observar posibles indicios de la presencia de un malware.	• Con las muestras Wannacry no fue posible ejecutar la aplicación de reconocimiento óptico de caracteres, ya que se cifra la memoria USB utilizada. • Por el contrario, con las muestras de TeslaCrypt y CryptoLocker, se efectuaron todas las fases. **Reconocimiento de patrones:** • WannaCry mayor número de coincidencias (17) y con Locky menor tiempo de ejecución (48,76 s). **Volcado de la memoria:** • CryptoLocker arrojó el mayor número de coincidencias (13) y el menor tiempo de ejecución lo arrojó TeslaCrypt con 2,05 s.

Nota: adaptado de Benavides y Roa (2018).

Tabla 3.

Caracterización para detección del malware ransomware (continuación)

Malware ransomware analizados	Características utilizadas para la detección del malware ransomware	Técnicas de detección y clasificación evaluadas	Proceso de detección y clasificación	Resultados de las métricas
CryptoLocker **WannaCry**	Para el analisis de las muestras de ransomware, Bazante (2019), se enfocó en las siguientes caracteristicas: • ID_ANALISIS: id del análisis realizado en cuckoo sandbox. • DURATION: tiempo de duración del análisis. • SCORE: calificación dada por cuckoo sandbox con referencia al ransomware usado en cada análisis. • REGKEY_WRITTEN: valor cuantificado de las claves de registro escritas por el ransomware. • REGKEY_OPENED: valor cuantificado de las claves de registro abiertas por el ransomware. • REGKEY_READ: valor cuantificado de las claves de registro leídas por el ransomware. • ARTEFACTO: etiqueta dada al tipo de ransomware usado en los análisis. • SO: nombre del sistema operativo que resultó víctima del ransomware.	• Árbol de decisión como modelo de predicción. • Algoritmo de validación cruzada. • Oracle VM VirtualBox. Software de virtualización con muestras alojadas en un repositorio de GITHUB. • Análisis estático	• Se implementa un ambiente de pruebas controlado y un análisis de muestras de ransomware, a través del filtrado de sus atributos, con los cuales se realiza una detección vía clasificación automática de ataques de ransomware creando un modelo de aprendizaje automático. • El ambiente controlado está conformado por un equipo principal (sistema operativo Ubuntu 16.04.1) y dos sistemas víctimas (sistema operativo Windows XP y 7). • Se realiza la instalación y configuración de cuckoo sandbox, con las herramientas adicionales requeridas para realizar los experimentos un modelo de aprendizaje automático. • Obtención de las muestras analizadas desde el repositorio de GITHUB y colocación de las mismas en la interfaz Web de Cuckoo. • Se suben los datos a la plataforma Rapidminer, para la visualización y organización de los datos que nutren el modelo de predicción. • Generación del modelo de aprendizaje automático con el algoritmo árbol de decisión, utilizando tres atributos: REGKEY_WRITTEN, REGKEY_OPENED, REGKEY_READ, para comprobar su efectividad. • Ejecución del modelo de aprendizaje automático y la obtención de los resultados, a través de un árbol de decisión; en una primera etapa con base en el atributo score y en una segunda etapa, solo utilizando las claves de registro de manera cuantificada y un algoritmo de validación cruzada.	• WannaCry es potencialmente más malicioso que CryptoLocker porque los valores obtenidos con los diferentes scores son mayores a 20 y tienen un promedio de 20.6 para Windows XP y 23.44 para Windows 7. • El número más alto de firmas activas para WannaCry es de 50, mientras que para CryptoLocker es de 16; siendo este último ransomware menos intrusivo • El modelo con el árbol de decisión arrojó una precisión del 47,50%, porque la cantidad de atributos filtrados es mayor. • El modelo con el algoritmo de validación cruzada arrojó una precisión igual a 97,50%; por lo tanto, este logra predecir el tipo de ransomware que atacó un determinado sistema.

Nota: adaptado de Bazante (2019).

Tabla 3.

Caracterización para detección del malware ransomware (continuación)

Malware ransomware analizados	Características utilizadas para la detección del malware ransomware	Técnicas de detección y clasificación evaluadas	Proceso de detección y clasificación	Resultados de las métricas
Petya **CryptoLocker** **WannaCry**	Las características tomadas en cuenta en este caso, según los aportes de Herrera et al. (2019), sin: • Artefacto: se refiere a los ransomware usados e identificados con los números 001 a 005. • Familia: se refiere a si el ransomware es de tipo encriptador o bloqueador. Se identifica por los números 1 y 2. • REGWRITE, REGOPEN REGREAD: se refiere a las claves de registro afectadas. • PROC: Se refiere a los procesos involucrados en la infección del ransomware. • PMFILES: se refiere al conjunto de archivos de volcado de memoria y los identificadores de procesos en memoria. Se han tomado estas características del objeto procmemory. • PMURLS: se refiere a las urls almacenadas en memoria durante el proceso de volcado. También se ha tomado del objeto procmemory. • NETHOSTS: se refiere a las direcciones IP de los hosts que interactuaron durante cada experimento realizado. Se han tomado del objeto network. • NETREQUEST: se refiere a direcciones de nombres de dominio que interactuaron o se hicieron llamadas durante el proceso de comunicación del artefacto. Se ha tomado del objeto network y la característica DNS de este.	• Volcados de memoria con el objeto Procmemory. • Visualización del comportamiento con el objeto Behavior. • Creación de una dataset con la herramienta Cuckoo.	• Creación de un ambiente controlado y virtualizado, así: máquina 1 contiene la herramienta Cuckoo Sandbox, víctima 1 con Windows XP y víctima 2 con Windows 7. • La máquina 2 sirvió para el procesamiento de los modelos a aplicar basados en la selección estadística de algoritmos de aprendizaje de máquina y la máquina 3 permitió el almacenamiento de grandes volúmenes de información (logs de sistemas) que se utilizaron con los modelos de aprendizaje de máquina.	Finalmente, se propuso y diseñó una dataset con atributos de los malware ransomware analizados.

Nota: adaptado de Herrera et al. (2019).

De acuerdo con la información de la tabla 3, es posible agregar que las técnicas de detección y clasificación de malware ransomware revelan importantes diferencias en relación con su utilización empírica, debido a la naturaleza, las métricas de evaluación y las características propias de estas técnicas y de este tipo de virus. En cuanto a las características de los ransomware se evidencia que las técnicas Máquina de Vector Soporte Monoclase (OC-SVM) y Naive Bayes y los paradigmas Virtualización de Funciones de Red (NFV) y las Redes Definidas por Software (SDN), utilizan en sus experimentos múltiples flujos de datos que comparten el mismo protocolo (TCP / UDP / ARP) en una ventana deslizante determinada; con la finalidad de hallar parámetros de patrones de tráfico de red limpio e infectado, según los escenarios elaborados al respecto y los cuales se van actualizando con los flujos recibidos durante los últimos 10s se utilizan para calcular las características agregadas.

En contraste, las diferencias en cuanto a las particularidades de las técnicas y métodos de Machine Learning empleadas con los malware ransomware también se identifican con relación a las técnicas de reconocimiento óptico de caracteres, OCR de la ventana emergente, reconocimiento de patrones y de volcado de la memoria RAM; dado que la detección y clasificación en este caso, se efectúa por medio de los elementos visuales de los virus analizados, como los caracteres que posee el archivo de texto usada para informar a la víctima la encriptación e infección de su sistema, junto con la imagen de la ventana emergente que aparece para notificar el cifrado de sus archivos. Las técnicas antes mencionadas tienen en común que son utilizadas para la identificación de las extensiones de estos virus, pero la detección propiamente dicha se realiza por medio de la identificación, visualización y análisis de los elementos gráficos que conforman el aviso de infección, rescate y procedimiento de recuperación.

Con respecto a las características utilizadas de los malware ransomware también se identificaron diferencias significativas, ya que con el uso de las técnicas Máquina de Vector Soporte Monoclase (OC-SVM) y Naive Bayes se emplean características de contexto (calculadas a partir de la ventana de flujos) y características locales (calculadas a partir del último flujo); en ambos casos, con datos numéricos y estadísticos. En divergencia con estas características, mediante las técnicas de reconocimiento óptico de caracteres, OCR de la ventana emergente, reconocimiento de patrones y de volcado de la memoria RAM, se emplean características clave como el Correo electrónico, URL, dirección de Bitcoin y dinero del rescate; según sus elementos gráficos y textuales, como el fondo de pantalla, la ventana emergente, los caracteres del mensaje de notificación y de la nota de rescate, el cifrado de la unidad de almacenamiento, entre otros.

Una tercera técnica basada en Machine Learning para la detección y predicción de malware ransomware, utilizada en la literatura, son los árboles de decisión, la cual fue utilizada para estudiar los virus de CrytoLockery y WannaCry por medio de la creación de ambientes controlados, tanto virtualizados como reales y que son utilizados para cargar las muestras infecciosas. Esta técnica de detección se apoya en una serie de herramientas en línea que permiten fortalecer el modelo de aprendizaje automático propuesto. Se destaca el uso de la herramienta Cuckoo Sandbox para la construcción de los escenarios de prueba y el cual, integrado con otras técnicas y modelos, se complementa para facilitar el desarrollo de procesos de detección y predicción (tabla 3).

Es relevante, señalar que el malware ransomware más analizado en la literatura para efectos de la presente investigación es WannaCry, porque según Maimó (2019) es uno de los virus más dañinos, maliciosos y que ha ganado relevancia en materia de ciberdelitos y ciberseguridad. Así mismo, se propaga automáticamente a través de la red aprovechando una vulnerabilidad en el bloque de mensajes del servidor basado en MS Windows (SMBv1); de ahí su interés por optimizar

los procesos de detección, clasificación, predicción y prevención frente a los ataques con este malware.

6.3. Construcción de la propuesta de la Técnica de Detección de Malware-Ransomware

A partir de los resultados referenciados en secciones anteriores, se continúa con la descripción y explicación de la técnica de detección de malware-ransomware que se propone con esta investigación, la cual está constituida por fases de desarrollo principales: 1) Fase de selección de los datos, 2) Fase de clasificación de malware y 3) Fase de detección del malware ransomware. Cada una de estas fases, a su vez, está compuesta por una serie de etapas consecuentes entre sí que permiten lograr la detección efectiva del malware ransomware, tal como se explica a continuación.

6.3.1. Fase 1. Selección de los datos.

Los algoritmos de aprendizaje automático que se articulan en la técnica propuesta necesitan de un set de datos que contenga atributos representativos del malware ransomware de manera que su clasificación de muestras no infectadas sea posible. Con la finalidad de generar el conjunto de datos requeridos para efectuar el respectivo entrenamiento y la validación al momento de la creación del sistema, y así mismo, para la extracción de los datos cuando ya el sistema esté en línea; es necesario plantear una etapa de minería y preprocesamiento de los datos, conforme con las características intrínsecas propias de este tipo de malware. Para efectos de esta propuesta, se determina que la fuente de los datos debe proporcionar información suficiente que permita discriminar los vectores de infección más utilizados por el ransomware: correos spear phishing con documentos adjuntos maliciosos, la explotación de vulnerabilidades (específicamente, ataques

de SQL Injetion, Cross Site Scripting, XSS y el MBR –Master Boot Record– del PC) y la explotación de aplicaciones web expuestas en internet.

En este sentido, para la minería de datos se requiere de la articulación de múltiples herramientas para la extracción de información desde las fuentes requeridas. Para la extracción de la información de los correos electrónicos se recomienda el uso de herramientas de footprinting; el footprinting es una técnica empleada en seguridad informática y también en hacking para recabar información acerca de las huellas de un sistema[29]; herramientas como Netcraft o la framework OSINT entre otras, brindan los recursos para generar reportes de: cabeceras, análisis de documentos adjuntos, breach data (violación de datos), formatos comunes de correo, listas negras, entre otras.

En cuanto a la recopilación de la información asociada con la explotación de vulnerabilidades, se puede utilizar un analizador de vulnerabilidades y un ejecutor de exploits, como Metasploit, que es un proyecto con código abierto para la seguridad informática, el cual además del análisis de vulnerabilidades, contribuye con la elaboración de una prueba de penetración "pentesting" y el desarrollo de firmas para sistemas de detección de intrusos (Palacios, 2015).

Así mismo, para la recopilación y extracción de la información relacionada con la explotación de aplicaciones web expuestas en internet, se puede utilizar herramientas de footprinting para extraer huellas DNS, los códigos fuentes, las traceroute (rutas de tráfico), un whois (reconocimiento) del dominio o IP solicitada. Alguna de las técnicas Web Use Mining, como la técnica de Reglas de Asociación o Association Rules, la técnica de Patrones de Secuencias

[29] Hay diversos tipos de Huellas: de motores de búsqueda, de red (sitios web, correos electrónicos, DNS), no técnica (de redes sociales).

o Sequential Patterns y la técnica Clustering o de Clasificación que se recomiendan en la literatura también pueden ser implementadas (Velasco, 2013).

Desde esta perspectiva y con el propósito de efectuar posteriormente una correcta extracción de los atributos a partir de los datos, es necesario construir una aplicación que genere la minería desde los reportes de las herramientas utilizadas y que se ejecute cuando se active un tipo de fuentes relacionada. Según los aportes de Benavides y Roa (2018), Herrera et al. (2019) y Bazante (2019) se definen las siguientes características específicas:

- Fuente de correo electrónico.
- URL.
- Pantallazo y OCR de ventanas emergentes para reconocimiento visual de nota de rescate.
- Tipo de extensión.
- Rastreo de registros del sistema
- REGKEY_WRITTEN: valor cuantificado de las claves de registro escritas por un ejecutable con firma desconcida (posible ransomware).
- REGKEY_OPENED: valor cuantificado de las claves de registro abiertas por un ejecutable con firma desconocida (posible ransomware).
- REGKEY_READ: valor cuantificado de las claves de registro leídas por un ejecutable con firma desconocida (posible acceso mal intencionado).
- Familia del ejecutable: permite reconocer si el ransomware es de tipo encriptador o bloqueador.
- Rastreo de PROC: Se refiere a los procesos involucrados en una ejecución, solo se desea si el fabricante o la firma es desconocido pudiendo ser la infección del ransomware.
- Las cabeceras del paquete.

- Empaquetado
- Código fuente (si se permite).
- Tipo de archivo adjunto.

Ahora bien, para la obtención de los datos de entrenamiento y validación, según las características indicadas y de acuerdo con los aportes de Maimó (2018), se considera la mitigación de los ransomware por medio de la Virtualización de Escritorios, Virtualización de Funciones de Red (NFV) y las Redes Definidas por Software (SDN); herramientas que en conjunto favorecen la adquisición de datos en panoramas de uso real, de forma natural por medio de las herramientas descritas, sobre los escritorios y los flujos de red.

Para el rastreo de los registros de navegación y los flujos de red, y siguiendo las recomendaciones de Maimó (2018), se define que, en tiempo real, con el tráfico de red generado por los dispositivos que conforman el ambiente controlado, se calculan los flujos de red agrupándolos en una ventana deslizante de tiempo equivalente a 10 segundos. En conjunto con el sistema de monitoreo de las características previamente determinadas, se propone un método de detección de anomalías, según las novedades en los patrones de tráfico (que posiblemente provengan de ataques de ransomware), mientras se generan las muestras, un clasificador simple etiquetará el tráfico que proviene de malware ransomware, esto es posible porque las fuentes de malware se obtienen de bases de datos abiertas con muestras pre-etiquetadas. Para la captura de información y de los paquetes de datos sobre la red, se recomienda el uso de una Sniffer, es decir, una aplicación especial para redes informáticas (un software) que se encarga de capturar y analizar

paquetes en tránsito (entrada y/o salida) en una red de comunicaciones entre dispositivos y que, en este caso, permitirá extraer las características de la ventana que se presentan a continuación:

- Media y desviación estándar de las duraciones de flujo.
- Media y desviación estándar de tiempo entre dos flujos consecutivos.
- Número de direcciones IP de destinos diferentes.
- Entropía de las direcciones IP de destino.
- Suma, máx., min., media, desviación estándar y mediana del total de paquetes.
- Suma, máx., min., media, desviación estándar y mediana de los paquetes fuente.
- Suma, máx., min., media, desviación estándar y mediana del total de bytes.
- Suma, máx., min., media, desviación estándar y mediana de los bytes de origen.
- Suma, máx., min., media, desviación estándar y mediana de la carga total.
- Suma, máx., min., media, desviación estándar y mediana de la carga de la fuente.
- % de puertos de origen / destino> 1024.
- % de puertos de origen / destino <1025.
- Número de puertos de origen y destino diferentes.
- Número de direcciones IP de destino, diferentes.
- Entropía de puertos de origen y destino.
- Entropía de la IP de destino.
- Mediana de la duración.
- Protocolo utilizado (TCP, UDP, ARP).
- Estado (INT, RST, FIN, CON).
- Ventana de tiempo usada para la generación del vector de características.

De los aportes de Herrera et al. (2019), se define el desarrollo de un ambiente controlado y virtualizado, para lo cual se propone una arquitectura cliente-servidor, conformada por lo menos con cuatro máquinas de estudio, discriminadas con los siguientes roles: una máquina principal (servidor) que contienen las herramientas de minería de datos requeridas por las máquinas cliente para la aplicación de la técnica propuesta, y algunas herramientas para la consolidación y procesamiento de las muestras consolidadas: Weka y Scikit-learn. Sumado a esto, el ambiente controlado y virtualizado, estaría constituido por dos clientes con las máquinas víctimas, cada una con características técnicas diferentes y una cuarta máquina propuesta para actuar como servidor en el almacenamiento de grandes volúmenes de información resultado de la minería y preprocesamiento. El set de datos resultado de la prueba se utiliza posteriormente para la clasificación y detección del malware.

En estas máquinas se deben desplegar aplicaciones web alojadas en el servidor y correos electrónicos con y sin vulnerabilidades (fuentes previamente etiquetadas); por lo tanto, los etiquetados serán ataque positivo (AP) y negativo (AN), una vez, se realice la limpieza de los datos. Así mismo, para la adquisición de las muestras de datos maliciosos con los ataques SQL Injection y Cross Site Scripting, se propone la utilización de dos sensores de vulnerabilidades: un firewall y un software de detector de intrusiones cuya información se registra en el disco del servidor de almacenamiento en formato plano contribuyendo a un único repositorio que se suma con el resultado de los sensores de tráfico HTTP, utilizados para la adquisición de datos normales (tráfico sin ataque). Los registros en el repositorio serán convertidos a un archivo CVS para su uso con las herramientas de minería de datos dentro de las librerías recomendadas para ciencia de datos como Weka y Scikit-learn. De igual forma, también se usarán bases de datos pre-etiquetadas obtenidas de diversos repositorios abiertos en línea, como GitHub, Gapminder; desde los cuales,

autores como por ejemplo Bazante y otros referenciados, obtuvieron los dataset utilizados para la experimentación.

Una vez se termine con la recopilación de las muestras, se procede con la limpieza de datos perdidos, la cual consiste en la identificación de los valores faltantes de la dataset, es decir, los datos que no pudieron obtenerse. Para la manipulación de los datos en la limpieza de valores perdidos, se recomiendan dos opciones al respecto: 1) una eliminación por lista, la cual implica eliminar toda la observación que tenga uno o varios datos perdidos, una alternativa que será aplicada con esta propuesta, siempre y cuando se trata de una cantidad pequeña de valores faltantes; y 2) imputación de valor central, en este caso, se recomienda reemplazar los datos perdidos con los valores de las medidas de tendencias centrales; para las variables categóricas es preferible utilizar la moda y para las variables numéricas, se propone aplicar la media.

Sumado al procedimiento de limpieza, si el dataset contiene datos nominales u ordinales, se prosigue con la codificación escalar de datos categóricos, con la finalidad de mejorar el rendimiento de las técnicas de ML que aquí se proponen mejorando los cálculos de distancia; un procedimiento que podría llevarse a cabo para la codificación de cadenas de texto complejas como cabeceras y códigos fuente, es la utilización de N-gramas y la aplicación Linguistic Inquiry and Word Count (LIWC), que combinan codificadores estadísticos clásicos con modelos de clustering basados en Machine Learning.

Para continuar, una vez se efectúa la codificación, se realiza el escalado de características, también llamado normalización de los datos, un procedimiento que se ejecuta para evitar o reducir anomalías de datos, redundancia y duplicaciones de características, lo que mejora notablemente la integración de los datos. Se propone que la normalización se realice en una escala gaussiana, mediante la combinación de los datos existentes por grupos, aclarando las relaciones lógicas entre

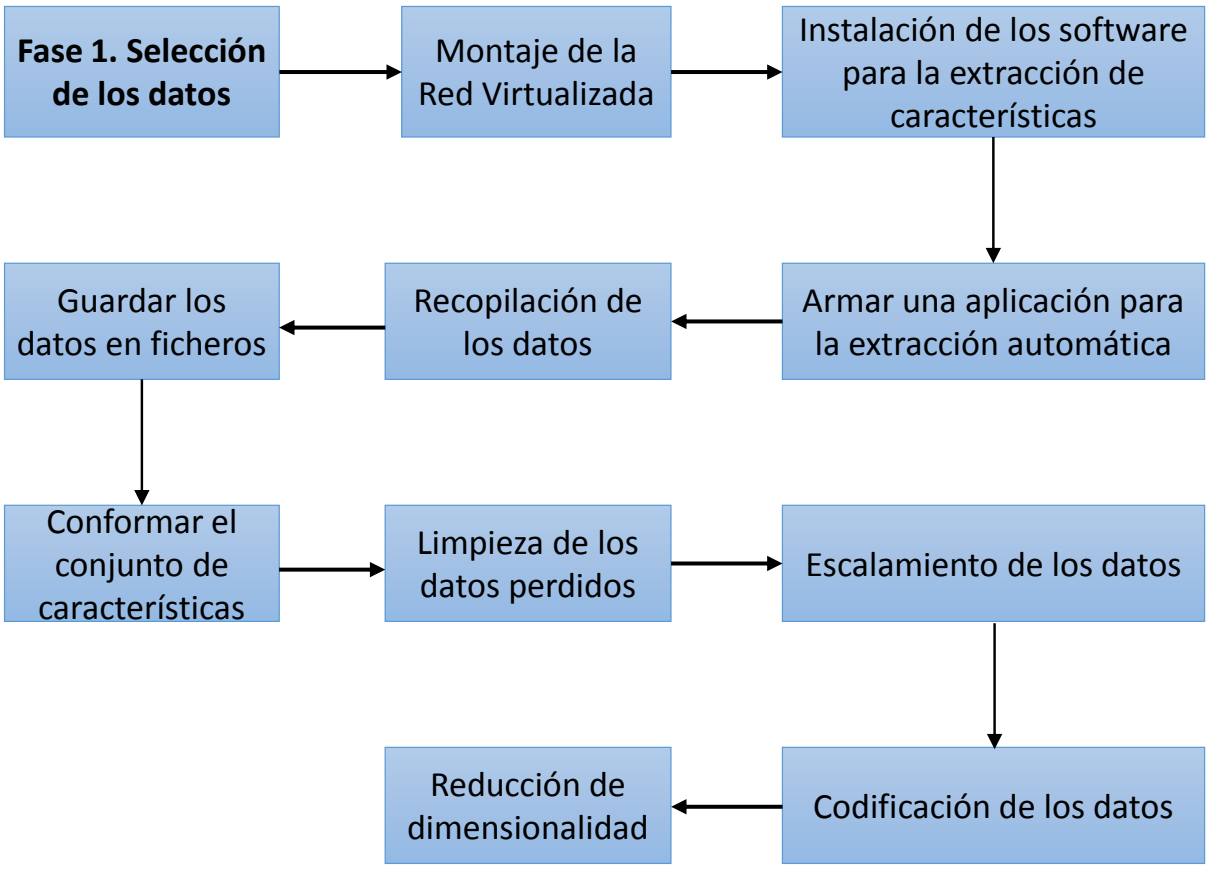
los grupos; dos acciones que contribuyen con la exactitud de los vínculos entre los campos del mismo tipo. Adicional a lo anterior, se procede con la reducción de dimensionalidad de los atributos, para lo cual se propone la aplicación de un Análisis de Componentes Principales (PCA), un método estadístico de la familia de técnicas de Machine Learning sin supervisión, con el cual no solo se eliminarán las características sin importancia o altamente correlacionadas (dependientes de otras), dejando los atributos críticos requeridos, según los modelos de clasificación y detección a utilizar, sino que además, favorece la extracción de información y garantiza la eficiencia óptima de los algoritmos mejorando el ritmo de entrenamiento y validación.

Al culminar el preprocesamiento de los datos y se obtiene el conjunto de datos limpios, se procede según los aportes de Radwan (2019), con la aplicación del método de división de prueba de tren, para la división de este conjunto, mediante la proporción del 70% de los datos para el entrenamiento del sistema y el 30% para la etapa de prueba y la evaluación de los algoritmos de clasificación y detección definidos al respecto.

En la Figura 10 se presenta un flujograma de la secuencia sugerida, desde sus etapas generales, para la recolección y preprocesamiento de los datos.

Figura 10.

Diagrama de flujo de la fase 1 de la técnica propuesta



Nota: elaboración propia.

6.3.2. Fase 2. Clasificación de malware

Cuando se ha efectuado el procesamiento de los datos, se continúa con la clasificación del malware ransomware, mediante la definición de los algoritmos de clasificación, los cuales, en este caso, corresponden específicamente con modelos de Machine Learning. Para esta técnica se determinó la utilización de múltiples clasificadores simples mezclados por Boosting: se propone el uso del algoritmo de Naïve Bayes (NB), porque de acuerdo con la literatura estudiada es de baja

complejidad (bajo uso de recursos) y presenta buenos resultados en la clasificación. De igual forma, el uso de Máquinas de Vector Soporte, dado que genera buenos resultados sin generar mucha carga en el entrenamiento y prueba. Se plantea el uso de un tercer algoritmo de Machine Learning, el Árbol de decisión (DT), algoritmo J48, que permitirá mejorar los resultados obtenidos, aumentando una capa de refinamiento con el objetivo de optimizar los resultados de clasificación de los primeros clasificadores y reducir las falencias en falsos positivos y falsos negativos. Este algoritmo obtiene altos niveles de rendimiento en función de su: recuperación (tasa de verdaderos positivos), tasa de falsos positivos, precisión (valor predictivo) y exactitud.

Para mayor redundancia, se propone el uso del algoritmo Random Forest, ya que además de generar buenos resultados de clasificación se trata de un algoritmo de ensamble, por lo tanto, permite efectuar un Boosting como técnica de validación cruzada. Este último algoritmo puede cargar al sistema en el momento del entrenamiento, sin embargo, por su naturaleza de combinar sus resultados, unos errores se compensan con otros y se puede obtener una predicción que generaliza mejor.

Ahora bien, la técnica agrupa 4 algoritmos que complementariamente debería garantizar una eficacia optima de clasificación, sin embargo, su adaptación es flexible, ya que solo en una etapa de implementación y prueba (que no está dentro del alcance de este trabajo), se podrá determinar si la redundancia de los algoritmos es necesaria (o en qué medida lo es) para lograr una clasificación optima e inteligente. El refinamiento de los hiperparámetros de cada algoritmo de clasificación se propone, mediante la elaboración de una matriz de confusión, teniendo en cuenta el número de instancias correctas e incorrectas de cada clase. TP (Verdadero Positivo) y TN (Verdadero Negativo) indican el número de instancias positivas y negativas que están clasificadas correctamente por el clasificador, mientras que FN (Falso Negativo) y FP (Falso Positivo) denotan

el número de instancias positivas y negativas clasificadas de forma incorrecta, respectivamente. A partir de los resultados obtenidos con la matriz de confusión, se procede con el cálculo de las métricas de evaluación de los clasificadores, tal como se plantea en la tabla 4.

Tabla 4.

Métricas para la evaluación de los clasificadores

Nombre de la métrica	Descripción de la métrica	Fórmula para su cáculo	Resultado óptimo
Precisión	La proporción de muestras de prueba predichas como positivas y correctas.	Precisión (P) = TP / (TP + FP).	Máquina de Vector Soporte = 99% Naïve Bayes (NB) = entre 99,48% Random Forest = 95% Árbol de decisión, con algoritmo J48 = 98%
Sensibilidad	La proporción de muestras de prueba que están etiquetadas como positivas y son correctas.	Sensibilidad (R) = TP / (TP + FN).	Máquina de Vector Soporte = 99% Naïve Bayes (NB) = 95% Random Forest = 98,39%. Árbol de decisión, con algoritmo J48 = 98%

Nota: elaboración propia.

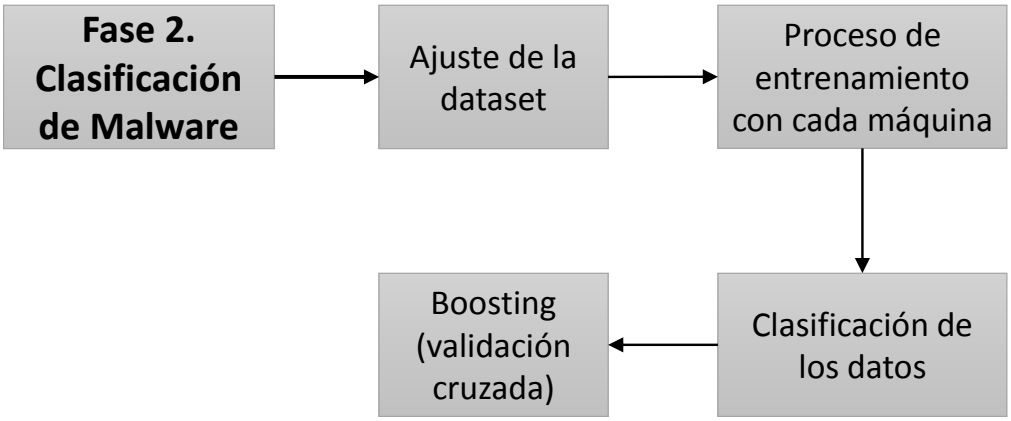
Se delimitan las métricas de eficacia a la precisión y la sensibilidad puesto que un porcentaje alto en ambas implica una reducción en FP y FN. La información, en cuanto al resultado óptimo, referenciada de la tabla 4, se obtuvo de la evaluación de los cuatro clasificadores referenciados en líneas previas desde la revisión de la literatura; obedece a la intencionalidad de que, a través de los experimentos que se efectúen en la etapa de implementación, la cual no corresponde con el alcance de esta investigación, permita comprobar los resultados reportados en la literatura, así como las virtudes de estos clasificadores. Es preciso clarificar que la cuantificación propuesta con las métricas de la tabla 3, corresponde con un promedio de los resultados de la

literatura, esto con el propósito de abarcar de forma integral la totalidad de puntuaciones porcentuales encontradas. Sin embargo, en una etapa de implementación las métricas tendrán que ser determinadas para sugerir una finalización en el afinamiento del sistema.

En la figura 11 se presenta el proceso construcción del sistema clasificador de detección desde el flujograma de las etapas de entrenamiento.

Figura 11.

Diagrama de flujo de la fase 2 de la técnica propuesta



Nota: elaboración propia.

6.3.3. Fase 3. Detección del malware ransomware y de evaluación

La implementación de los clasificadores seleccionados en la fase anterior tendrá que afinarse mediante los experimentos necesarios para alcanzar las métricas definidas con dichas técnicas. Por lo tanto, para optimizar los resultados de estas métricas, los parámetros de configuración que se sugieren para construir los clasificadores y para el ajuste de estos, son los siguientes:

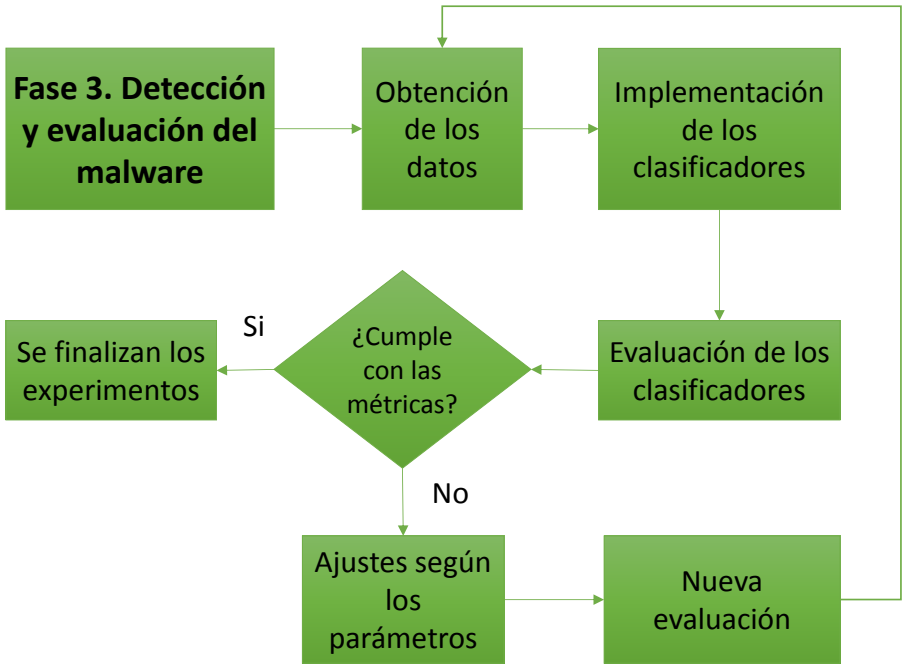
- **Criterios del árbol de decisión, algoritmo J48:** en este caso, se debe tener en cuenta la relación de ganancia, una medida porcentual que hace referencia a la desorganización de un sistema, denominado entropía; con el que se busca agregar más información a los nodos más impuros del árbol de decisión y reducir información a los nodos menos impuros. Se propone una profundidad máxima de 10 niveles bajo el nodo raíz, cantidad suficiente para aprender relaciones más específicas y para efectuar los ajustes requeridos durante el Boosting.
- **Random Forest:** con respecto a esta técnica los parámetros que se proponen están relacionados con el número de árboles, los cuales deben ser de mínimo 100 porque es una cantidad suficiente para estabilizar el error, ya que una cantidad inferior o superior puede ser ineficiente. De nueva cuenta Se propone una profundidad mínima de 2 y máxima de 10 niveles bajo el nodo raíz, cantidad suficiente para aprender relaciones más específicas y para efectuar los ajustes requeridos durante el Boosting.
- **Naïve Bayes (NB):** en cuanto a esta técnica, los parámetros de ajustes consideran una corrección del modelo Naïve Bayes utilizando el argumento Laplace=1, para incrementar e incorporar las probabilidades de casos raros y obtener un resultado verdadero.
- **Maquinas Vector Soporte (SVM):** en el caso de esta técnica, los ajustes se efectúan con el parámetro de aumento de la dimensión mediante el cálculo de un kernel polimórfico una función que represente mejor la múltiple dimensionalidad de los atributos de entrada para optimizar la separación entre los vectores de soporte de las clases (C en rango 10 a100 y Lamda mayor a 0,2), y así lograr la mejor separación en el espacio original.

Conforme con lo anterior, es pertinente indicar que para la evaluación se tomará el conjunto de los datos de prueba (30%), se introducirán en la maquina entrenada en la fase anterior, y se comprobarán los resultados de detección a partir de las métricas definidas en esta propuesta. Si los resultados coinciden con las métricas, se finalizan los experimentos; pero si los resultados no cumplen con las métricas previamente determinadas, se realizan los ajustes requeridos sobre los hiperparámetros indicados anteriormente y se inicia de nuevo la evaluación, hasta que se finalmente los resultados coincidan con las métricas.

En la figura 12 se presenta el proceso de evaluación del sistema desde el flujograma de las etapas de validación.

Figura 12.

Diagrama de flujo de la fase 3 de la técnica propuesta



Nota: elaboración propia.

7. Resultados y Discusión

En el desarrollo de la presente investigación se evidenció que el malware ransomware continúa siendo una amenaza en potencia para los sistemas de información del sector empresarial y en general para cualquier usuario de un sistema de cómputo conectado a la web; no solo por la propagación de software maliciosos más novedosos, infecciosos y destructivos, sino también por los efectos que representan y producen en el desarrollo de las actividades de sus víctimas. Una inefectiva ciberseguridad en los equipos y las redes, además de la poca información sobre los riesgos derivados de la navegación ingenua por la red, deriva en brechas que aprovechan los cibercriminales para la entrada de software malicioso – ransomware que captura la información confidencial. Tanto los responsables directos de los equipos (sean personas u organizaciones), como también las afectaciones indirectas derivadas de la filtración de información deja expuesta información confidencial de sus usuarios alrededor del mundo.

Ante esta realidad, se hace cada vez más indispensable la consecución e implementación de modelos eficaces para la detección efectiva del malware ransomware, encontrando en las técnicas de Machine Learning una alternativa acorde con las características intrínsecas propias de este malware y que reporta altos niveles de rendimiento, en cuanto precisión y exactitud y también en relación con la posibilidad de integrar diferentes tipos de técnicas de aprendizaje automático, con la finalidad de incrementar y potenciar los resultados obtenidos en materia de detección y clasificación. El objetivo de una detección temprana (antes de que se ejecute) es evitar: la encriptación o bloqueo de los documentos y del sistema de las víctimas y el consecuente pago del rescate solicitado por los ciberdelincuentes que lanzan el malware ransomware.

Si bien desde la literatura, se identificó la importancia que está teniendo la aplicación de las técnicas de Machine Learning para la detección y clasificación de diversas familias de malware, también se evidenció que los antecedentes acerca de su implementación en relación con el malware ransomware siguen siendo muy limitados; esto debido a que sus modalidades y técnicas de acceso fácilmente se resguardan en archivos o enlaces que en numerosos casos no considerados como aplicaciones maliciosas, porque suelen pasar como documentos ordinarios que parecen enviados de empresas o personas conocidas. De igual forma, los procesos de detección y clasificación se dificultan porque en determinados casos, el malware ransomware también se suele activar en un segundo plano, mientras se efectúan otras acciones, desviando así, la atención de los usuarios del sistema; de acuerdo con Ceballos et al. (2019), el software malicioso se ha convertido en un delito que ha incrementado en el mundo en un 767%, principalmente, por los bajos niveles de ciberseguridad de sus víctimas.

Ahora bien, aunque la literatura hallada con relación a la implementación de técnicas de aprendizaje automático específicamente para la detección y clasificación de malware ransomware es limitada; es relevante señalar que autores como Vivanco-Toala et al. (2020), Maimó (2019), Bazante (2019), Herrera et al. (2019), Benavides y Roa (2018), identificaron la magnitud que ha alcanzado la propagación del malware ransomware en el contexto nacional e internacional, debido a la multivariedad de ataques, técnicas de acceso y familias que conforman este tipo de amenazas informáticas.

Así mismo, estos autores resaltan las bondades de las técnicas de Machine Learning para la ejecución de acciones de detección y clasificación del malware ransomware, porque ofrecen un nivel de rendimiento óptimo y buen comportamiento en detectar ransomware desconocido, como se evidenció con el modelo Naive Bayes (Maimó, 2019); y además, contribuyen con el

reconocimiento de patrones que corresponde a una de las muestras de ransomware almacenadas previamente y que están involucradas en un ataque informático, tal como se logra con la aplicación del Reconocimiento Óptico de Caracteres (OCR), el volcado de la memoria RAM y la extracción de archivos relevantes (Benavides & Roa, 2018). De igual forma, la aplicación del árbol de decisión para el entrenamiento y los operadores Apply Model y Perfomance para la prueba, junto con un algoritmo de validación cruzada, permiten predecir el tipo de ransomware que ataca un determinado sistema (Bazante, 2019).

Con respecto a la detección y clasificación de diversas familias de malware, se destaca la implementación de nuevos y mejores algoritmos clasificadores que están demostrando niveles de rendimiento significativos, como se ha presentado en años recientes con el algoritmo árbol de decisión J48, ya que en la etapa de entrenamiento se mejoran las métricas como la precisión y la exactitud (Rodríguez, 2018); sumado al rendimiento este modelo en función de su recuperación (tasa de verdaderos positivos), tasa de falsos positivos, precisión (valor predictivo) y exactitud (Firdausi et al., 2010). Así mismo, el árbol de decisión J48, es un algoritmo que ha arrojado un alto porcentaje de clasificaciones correctas y mayor rapidez en cuanto al tiempo empleado para verificar el modelo con un conjunto de datos test determinado, especialmente, cuando es implementado con la librería Scikit-learn (Moscardó, 2018); porque se suele guiar por múltiples atributos de los cuales difícilmente un atacante tendrá control total, y que, por lo tanto, siempre sería detectado (Romero, 2019).

Es relevante señalar que la capacidad y precision de los modelos utilizados desde la literatura para la detección y clasificación del malware ransomware, depende directamente del método utilizado, el número de atributos, el conjunto de datos y los registros del conjunto de datos, las técnicas de preprocesamiento y las herramientas implementadas en el modelo (Al-Janabi &

Altamimi, 2020); y así mismo, está influenciado por los parámetros asociados a los paquetes de entrada y salida, el comportamiento y procesamiento de CPU (Herrera et al. 2019).

Así las cosas, de acuerdo con los resultados obtenidos con el desarrollo de la presente investigación, es posible determinar que la técnica propuesta es de carácter integradora, dado que conjuga y potencia las capacidades individuales de cada uno de los métodos propuestos, sumado a la utilización de las mejores técnicas de preprocesamiento aprovechables a través de internet, como es el caso de Weka y Scikit-learn, por ejemplo; las cuales ponen a disposición de la comunidad académica y científica diversas bases de datos y herramientas para la implementación de técnicas de Machine Learning que resulten en una variedad de documentos investigativos que den cuenta de los procedimientos y hallazgos encontrados con el uso de estas herramientas; los mismos que pueden ser recopilados y mejorados por otros estudiantes y autores.

Se trata de una propuesta que si bien no fue implementada por limitaciones de tiempo y recursos del estudiante investigador, contribuye con el desarrollo de las primeras etapas implicadas con el proceso de detección y clasificación del malware ransomware, convirtiéndose en una oportunidad de trabajo futuro, que se espera sea aplicado en la realidad actual, el cual puede ser optimizado con la utilización del lenguaje de programación Python, gracias a su versatilidad multiplataforma y multiparadigma, a la legibilidad, potencia y limpieza de su código; el cual además, de tener licencia abierta, trabaja con una extensa biblioteca de herramientas y con librerías de Scikit-learn enfocadas especialmente, en el aprendizaje automático.

Sumado a esto, se recomienda la implementación de esta técnica teniendo en cuenta los ataques de ransomware dirigido, los cuales se realizan contra una víctima elegida con el objetivo de extorsionar, lo que se presenta en mayor medida, con respecto a las organizaciones nacionales e internacionales, de diferentes sectores (Data Center Market, 2021). Así mismo, aunque la técnica

de Reconocimiento Óptico de Caracteres no se tuvo en cuenta con esta propuesta más allá que para la extracción de características, se recomienda el desarrollo de una nueva alternativa que la integre con otras técnicas dentro del proceso de clasificación, con la finalidad de lograr una identificación más compleja de los patrones que caracterizan el malware ransomware, en especial, con respecto a los elementos gráficos que lo componen.

Con la técnica propuesta, resultado de este proyecto de investigación, se responde al objetivo general y se confirma la hipótesis formulada con este proyecto, es decir, se determinó que la técnica propuesta ofrece otra posibilidad de acercamiento al desarrollo de una medida preventiva del malware ransomware, a través del análisis y el desglosamiento de sus formas de intervención, las cuales se aprovechan para el entrenamiento de nuevos clasificadores y para las pruebas requeridas para la detección de este tipo de virus en el momento de su implementación.

8. Impacto Esperado

Según los resultados obtenidos con la presente investigación, se espera un impacto positivo, en el ámbito nacional e internacional, con respecto a la detección frente a posibles y futuros ataques informáticos, con los malware ransomware; porque se propone una técnica que integra las características potenciales y más destacadas de los modelos de detección y clasificación de este tipo de virus, referenciados en los documentos revisados y analizados para cumplir con los objetivos formulados con este estudio.

De igual forma, se espera que la técnica propuesta sea implementada, inicialmente, en el contexto académico, con la finalidad de corroborar cada una de las indicaciones dadas en las respectivas fases que componen dicho modelo y de incentivar nuevos procesos investigativos que, no solo permitan la consecución de las métricas establecidas con esta técnica, sino también la introducción de posibles mejoras a la misma.

Adicionalmente, se pretende que la ejecución y puesta en marcha de esta propuesta, se refleje en mejores niveles de ciberseguridad para las organizaciones de hoy y los usuarios, beneficiándose con una herramienta efectiva de prevención del malware ransomware y de protección con respecto a la confidencialidad de su información.

9. Conclusiones

De conformidad con los objetivos planteados con la presente investigación, se concluye que la detección y clasificación del malware ransomware, durante el período comprendido entre los años 2016 y 2020; se ha optimizado considerablemente, mediante el desarrollo y la implementación de una multivariedad de técnicas de Machine Learning, principalmente, según los resultados obtenidos con este proyecto y la revisión de 35 documentos, con respecto a las técnicas enfocadas en específico en el malware ransomware, tales como: Máquinas de Vector Soporte (SVM) (lineal y no lineal) (14%), Naïve Bayes Classifier (13%) y Random Forest (bosque aleatorio) (12%); debido a los altos niveles de precisión y exactitud que han arrojado en los experimentos efectuados por autores como Maimó (2019), Bazante (2019) y Benavides y Roa (2018), especialmente; teniendo en cuenta que estos investigadores demostraron, las capacidades en la utilización de estas técnicas mediante procedimientos integradores, es decir, aplicando múltiples técnicas para complementar los resultados obtenidos por estas de forma independiente.

Aunado a lo anterior, se concluye además, que características intrínsecas de los malware ransomware como los patrones de ataque, acceso y encriptación de los archivos del sistema de la víctimas y creciente número de variedades del software malicioso, son cada vez más sofisticados y potentes, lo que no solo afecta en mayor medida los sistemas informáticos de las víctimas, sino que también dificulta su detección y clasificación temprana, por medio de los antivirus utilizados por numerosas organizaciones y usuarios en la actualidad para la protección de su ciberseguridad; de ahí que el llamado enérgico por parte de las plataformas en red, este dirigido hacia el desarrollo de nuevos y mejores modelos de protección, basados en técnicas de aprendizaje automático que

obtengan la capacidad de identificar los complejos patrones inherentes a este tipo de virus, según el establecimiento de una serie parámetros clave.

Con respecto a la técnica de prevención propuesta con este proyecto, se determina que es una alternativa integradora que aprovecha las capacidades potenciales de las técnicas más destacadas y utilizadas previamente por otros autores, en especial por Maimó (2019) y Bazante (2019); quienes a través de los experimentos efectuados obtuvieron resultados muy buenos resultados (por encima del 95%) en materia de precisión y exactitud de las mismas, conforme con el uso de las diferentes herramientas disponibles en la red, tanto para el preprocesamiento de los datos como para las fases de clasificación, detección y evaluación del malware ransomware.

10. Recomendaciones Futuras

En primer lugar, para fortalecer los resultados de la presente investigación, se recomienda la implementación de la técnica de prevención propuesta con este proyecto, conforme con las acciones y procedimientos planteados con el mismo; proceso que inicialmente, es conveniente que sea efectuado en el ámbito académico-educativo, con el propósito de obtener las métricas de evaluación determinadas con esta propuesta y de este modo, establecer las posibilidades de ejecución en otros contextos mayores.

En segundo lugar, a partir de la información plasma en este estudio, pueden surgir nuevas investigaciones al respecto, enfocadas especialmente, para nutrir la literatura utilizadas en la consecución de los objetivos formulados con el mismo y la actualización de los parámetros establecidos con la técnica propuesta; teniendo en cuenta que debido a la magnitud que han representado los ataques con el malware ransomware seguirán surgiendo nuevas y mejores técnicas y más autores interesados en el estudio de esta tema.

Referencias

Al-Janabi, M., & Altamimi, A. (2020). A Comparative Analysis of Machine Learning Techniques for Classification and Detection of Malware. *21st International Arab Conference on Information Technology (ACIT)*, 1-9. Obtenido de https://ieeexplore.ieee.org/abstract/document/9300081

Asad, A., Mansur, R., Zawad, S., Evan, N., & Hossain, M. (2020). Analysis of Malware Prediction Based on Infection Rate Using Machine Learning Techniques. *IEEE Region 10 Symposium (TENSYMP)* (págs. 706-709). Dhaka, Bangladesh: IEEE. Obtenido de https://ieeexplore.ieee.org/abstract/document/9230624

Aver, H. (10 de noviembre de 2020). *El ransomware en el 2020.* Obtenido de www.kaspersky.es: https://www.kaspersky.es/blog/ransomware-incidents-2020/24211/

Bazante, F. (2019). *Análisis de correlación automática para detección de ataques ransomware en ambiente de pruebas.* Quito: [Trabajo de Titulación Previo a la Obtención del Título de Ingeniero en Sistemas Informáticos y de Computación]. Escuela Politécnica Nacional. Facultad de Ingeniería de Sistemas. Obtenido de https://bibdigital.epn.edu.ec/handle/15000/20121?mode=simple

Becerra, C., & Vargas, J. (2019). *Análisis de la eficacia de Machine Learning para la identificación de Phishing y Spam.* Bogotá D.C., Colombia: Universidad de los Andes. Departamento de Ingeniería de Sistemas y Computación. Maestría en Seguridad de la Información. Obtenido de https://proyectosmaestrias.virtual.uniandes.edu.co/images/v3FfQb2K0RghJ82V7UP8zATZLZDNImE6xyXtCKud.pdf

Benavides, L., & Roa, C. (2018). *Herramienta de Extracción de Información de Malware.* Madrid: [Trabajo Fin de Grado]. Universidad Complutense de Madrid. Facultad de Informática. Grado de Ingeniería Informática. Obtenido de https://eprints.ucm.es/id/eprint/56247/1/069.pdf

Carmona, E. (2016). Tutorial sobre Máquinas de Vector Soporte (SVM). *Researchgate.net*, 1-27. Obtenido de https://www.researchgate.net/publication/263817587_Tutorial_sobre_Maquinas_de_Vectores_Soporte_SVM

Ceballos, A., Bautista, F., Mesa, L., & Argáez, C. (2019). *Informe de las tendencias del Cibercrimen en Colombia (2019-2020).* Bogotá D.C.: Cámara Colombiana de Informática y Telecomunicaciones (CCIT) & Policía Nacional. Obtenido de https://www.ccit.org.co/wp-content/uploads/informe-tendencias-cibercrimen_compressed-3.pdf

Chamorro, A. (2020). *Malware Detection with Machine Learning.* Barcelona: [Trabajo Fin de Grado en Enginyeria Informatica], Universitat Autònoma De Barcelona (UAB). Escola D'enginyeria (EE). Obtenido de https://ddd.uab.cat/pub/tfg/2020/tfg_285427/MalwareDetection-Informe-final.pdf

Choudhary, S., & Sharma, A. (2020). Malware Detection & Classification using Machine Learning. *International Conference on Emerging Trends in Communication, Control and Computing (ICONC3)*, 1-4. Obtenido de https://ieeexplore.ieee.org/abstract/document/9117547

Cisco Systems, Inc. (s.f.). *What Is a Firewall?* Obtenido de www.cisco.com: https://www.cisco.com/c/en/us/products/security/firewalls/what-is-a-firewall.html

Cusaria, C., & Fagua, A. (2017). Estrategias de Ingeniería Social a Partir del Análisis de Datos Residuales en la Ciudad de Tunja. *Revista Ciencia, Innovación y Tecnología (RCIY), 3*, 39-50. Obtenido de https://www.jdc.edu.co/revistas/index.php/rciyt/article/view/73/69

Darus, F., Salleh, N., & Ariffin, A. (2018). Android Malware Detection Using Machine Learning on Image Patterns. *Cyber Resilience Conference (CRC)*, 1-2. Obtenido de https://ieeexplore.ieee.org/abstract/document/8626828

Data Center Market. (27 de abril de 2021). Los ataques de ransomware dirigido crecieron un 767% en 2020. *Computing*. Obtenido de https://www.computing.es/seguridad/noticias/1125265002501/ataques-de-ransomware-dirigido-crecieron-767-2020.1.html

ESET Latinoamérica. (noviembre de 2015). *Guía de respuesta a una infección por malware.* Obtenido de www.welivesecurity.com: https://www.welivesecurity.com/wp-content/uploads/2015/11/Guia_respuesta_infeccion_malware_ESET.pdf

Estrada, C. (2018). *Estudio sobre el malware Ransomware.* Cataluña: [Trabajo Final de Máster para obtener el título de Máster Interuniversitario de Seguridad de las Tecnologías de la Información de las Comunicaciones]. Instituto Nacional de Ciberseguridad. Obtenido de http://openaccess.uoc.edu/webapps/o2/bitstream/10609/89025/6/cestradacolTFM0119memoria.pdf

Fernández, Y. (2 de junio de 2020). *¿Cuál es la diferencia: malware, virus, gusanos, spyware, troyanos, ransomware, etcétera?* Obtenido de www.xataka.com: https://www.xataka.com/basics/cual-es-la-diferencia-malware-virus-gusanos-spyware-troyanos-ransomware-

etcetera?utm_source=whatsapp_AMP&utm_medium=social&utm_campaign=botonerammobile_AMP

Firdausi, I., Lim, C., Erwin, A., & Nugroho, A. (2010). Analysis of Machine learning Techniques Used in Behavior-Based Malware Detection. *Second International Conference on Advances in Computing, Control, and Telecommunication Technologies*, 201-203. Obtenido de https://ieeexplore.ieee.org/abstract/document/5675808/

Flores, J. (28 de junio de 2019). *Qué es el 5G y cómo nos cambiará la vida.* Obtenido de National Geographic: https://www.nationalgeographic.com.es/ciencia/que-es-5g-y-como-nos-cambiara-vida_14449

García, Á. (21 de Enero de 2020). *Auditoría automatizada basada en un sistema de detección de vulnerabilidades y en la explotación controlada de amenazas software.* Málaga: Universidad de Málaga. Escuela Técnica Superior de Ingeniería Informática. Departamento de Lenguajes y Ciencias de la Computación. Obtenido de https://riuma.uma.es/xmlui/bitstream/handle/10630/19202/GarciafernandezalvaroMemoria.pdf?sequence=1&isAllowed=y

Godoy, A. (2017). Técnicas de aprendizaje de máquina utilizadas para la minería de texto. *Revista de Investigación Bibliotecológica*, *31*(71), 103-126. Obtenido de http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S0187-358X2017000100103

González, L., & Vázquez, R. (julio-diciembre de 2015). Clasificación de Malware mediante Redes Neuronales Artificiales. *Revista del Centro de Investigación, 11*(44), 69-102. Obtenido de https://www.redalyc.org/articulo.oa?id=34242142004

Hernández, R., Fernández, C., & Baptista, M. (2014). *Metodología de la Investigación* (6 ed.). México D.F.: McGraw-Hill / Interamericana Editores, S.A. De C.V.

Herrera, J., Bazante, F., Barona, L., Valdivieso, Á., & Hernández-Álvarez, M. (octubre de 2019). Dataset de Ransomware basado en análisis dinámico. *RISTI: Revista Ibérica de Sistemas e Tecnologias de Informação*, 248-261. Obtenido de https://www.researchgate.net/profile/Juan-A-Herrera-Silva/publication/337652325_Dataset_de_Ransomware_basado_en_analisis_dinamico/links/5de29979299bf10bc334ea21/Dataset-de-Ransomware-basado-en-analisis-dinamico.pdf

Herrero, R. (2018). *Seguridad de redes y aplicaciones distribuidas. Programas maliciosos, antivirus y uso de emuladores de CPU en técnicas de análisis de malware.* Cataluña: [Memoria del Trabajo de Fin Máster]. Universitat Oberta de Catalunya (UOC). Máster Internuniversitario en Seguridad de las Tecnologías de la Información y de las Comunicaciones. Obtenido de http://openaccess.uoc.edu/webapps/o2/bitstream/10609/91026/6/rafaherreroTFM1218memoria.pdf

Kaspersky Lab. (2018). *Kaspersky Security Bulletin: Historia del año 2017.* Obtenido de https://media.kasperskycontenthub.com/wp-content/uploads/sites/63/2018/03/12102102/KSB_Story_of_the_Year_Ransomware_FINAL_ES.pdf

Kaspersky Lab. (2019). *Boletín de seguridad Kaspersky 2018. La historia del año: el malware de criptominería.* Obtenido de https://media.kasperskycontenthub.com/wp-content/uploads/sites/63/2018/12/05060830/KSB2018_Story-of-the-year_Miners_SP.pdf

Kim, J.-W., Namgung, J., Moon, Y.-S., & Choi, M.-J. (2020). Experimental Comparison of Machine Learning Models in Malware Packing Detection. *21st Asia-Pacific Network Operations and Management Symposium (APNOMS)*, (págs. 377-380). Asia. Obtenido de https://ieeexplore.ieee.org/abstract/document/9237007/

Kundro, D. (29 de julio de 2020). *Análisis del código fuente de un ransomware escrito en Python.* Obtenido de www.welivesecurity.com: https://www.welivesecurity.com/la-es/2020/07/29/analisis-codigo-fuente-ransomware-escrito-python/

Lazzeri, F. (12 de abril de 2021). *Aprendizaje profundo frente a aprendizaje automático en Azure Machine Learning*. Obtenido de docs.microsoft.com: https://docs.microsoft.com/es-es/azure/machine-learning/concept-deep-learning-vs-machine-learning

Lu, S., Ying, L., Lin, W., Wang, Y., Nie, M., Shen, K., . . . Duan, H. (2019). New Era of Deep Learning-Based Malware Intrusion Detection: The Malware Detection and Prediction Based On Deep Learning. *Computer Science*, 1-30. Obtenido de https://arxiv.org/pdf/1907.08356.pdf

Maimó, L. (15 de Julio de 2019). *Detección de bonets y ransomware en redes de datos mediante técnicas de aprendizaje automático.* Murcia: Universidad de Murcia. Escuela Internacional de Doctorado. Facultad de Informática. Obtenido de https://digitum.um.es/digitum/bitstream/10201/73765/1/Lorenzo%20Fern%c3%a1ndez%20Maim%c3%b3%20Tesis%20Doctoral%20s%20Art.pdf

Maimó, L., Gómez, Á., Clemente, F., Pérez, M., & Pérez, G. (2018). A Self-Adaptive Deep Learning-Based System for Anomaly Detection in 5G Networks. *Journal IEEE Access*(6), 1-12. Obtenido de https://www.researchgate.net/profile/Manuel-Perez-62/publication/323000892_A_Self-Adaptive_Deep_Learning-

Based_System_for_Anomaly_Detection_in_5G_Networks/links/5a99c32da6fdcc3cbac92dc0/A-Self-Adaptive-Deep-Learning-Based-System-for-Anomaly-Detection-in-5

Management Solutions. (2018). *Machine Learning: una pieza clave en la transformación de los modelos de negocio.* España. Obtenido de https://www.managementsolutions.com/sites/default/files/publicaciones/esp/machine-learning.pdf

Márquez, J. (2017). Armas Cibernéticas. Malware Inteligente para Ataques Dirigidos. *Revista Ingenierías USBMed*, *8*(2), 48-57. Obtenido de https://dialnet.unirioja.es/descarga/articulo/6071434.pdf

Mayorga, G. (2017). *Uso de analíticas para predecir los computadores afectados por malware, en una institución financiera en Colombia, 2017.* Bogotá D.C.: Escuela Colombiana de Ingeniería. Maestría de Gestión de Información. Obtenido de https://docplayer.es/57436977-Uso-de-analiticas-para-predecir-los-computadores-afectados-por-malware-en-una-institucion-financiera-en-colombia-autor-gerardo-mayorga-garcia.html

Moscardó, J. (2018). *Aprendizaje supervisado para la detección de amenazas Web.* Cataluña: [Trabajo Fin de Máster]. Universitat Oberta de Catalunya. Maestría en Seguridad de las TIC. Obtenido de http://openaccess.uoc.edu/webapps/o2/bitstream/10609/91066/6/jmoscardoTFM0119memoria.pdf

Mundaca, R. (7 de julio de 2020). *Malware y otros "ware" que te hacen sufrir y cómo debes cuidarte.* Obtenido de tecnologias.uchile.cl: https://tecnologias.uchile.cl/malware-y-otros-ware/

Nivaashini, M., Soundariya, R., Vidhya, H., & Thangaraj, P. (2018). Comparative Analysis of Feature Selection Methods and Machine Learning Algorithms in Permission based Android Malware Detection. *2018 International Conference on Intelligent Computing and Communication for Smart World (I2C2SW)*, (págs. 72-77). Erode, India. Obtenido de https://ieeexplore.ieee.org/abstract/document/8997527/

Organización Deloitte. (12 de febrero de 2021). *Las 10 tendencias en ciberseguridad que marcarán el 2021*. Obtenido de www2.deloitte.com: https://www2.deloitte.com/cl/es/pages/risk/articles/diez-tendencias-ciberseguridad-2021.html#

Padilla, M. (2010). Antivirus: una herramienta indispensable para nuestra seguridad. *Revista Seguridad*(4), 1-6. Obtenido de https://revista.seguridad.unam.mx/printpdf/2103

Paez, S. (2020). *Tema: Seguridad Informática.* Obtenido de www.goconqr.com: https://www.goconqr.com/c/87914/course_modules/137217-tema--seguridad-inform-tica#

Palacios, J. (2015). *Análisis de Vulnerabilidades de una Red Corporativa mediante Herramientas de Descubrimiento Activas.* Sevilla, España: [Trabajo Fin de Grado]. Universidad de Sevilla. Escuela Técnica Superior de Ingeniería. Departamento de Ingeniería Telemática. Grado en Ingeniería de las Tecnologías de Telecomunicación. Obtenido de http://bibing.us.es/proyectos/abreproy/90522/fichero/Memoria+del+Trabajo+Fin+de+Grado.pdf

Radwan, A. (2019). Machine Learning Techniques to Detect Maliciousness of Portable Executable Files. *International Conference on Promising Electronic Technologies (ICPET)*, 86-90. Obtenido de https://ieeexplore.ieee.org/abstract/document/8925324

Ricoy, C. (2006). Contribución sobre los paradigmas de investigación. *Educação. Revista do Centro de Educação, 31*(1), 11-22. Obtenido de https://www.redalyc.org/pdf/1171/117117257002.pdf

Rodríguez, J. (2018). *Aplicación de técnicas de Machine Learning a la detección de ataques.* Cataluña: [Trabajo de Fin de Máster]. Universitat Oberta de Catalunya. Maestría en Seguridad de las TIC (MISTIC). Obtenido de http://openaccess.uoc.edu/webapps/o2/bitstream/10609/81126/11/jmrodriguez85TFM0618memoria.pdf

Romero, A. (2019). *Clasificación de flujos de tráfico en Internet utilizando técnicas de aprendizaje automático.* Madrid: [Trabajo Fin de Grado en Ingeniería de Tecnologías y Servicios de Telecomunicación]. Universidad Autónoma de Madrid. Escuela Politécnica Superior. Obtenido de https://repositorio.uam.es/bitstream/handle/10486/689041/romero_del_campo_alejandro_tfg.pdf?sequence=1&isAllowed=y

Romero, M., Figueroa, G., Vera, D., Álava, J., Parrales, G., Álava, C., . . . Castillo, M. (2018). *Introducción a la seguridad informática y el análisis de vulnerabilidades.* Alicante: Área de Innovación y Desarrollo, S.L. Obtenido de https://www.3ciencias.com/wp-content/uploads/2018/10/Seguridad-inform%C3%A1tica.pdf

Ruiz, J. (2019). *Detección de Malware, Métodos Estadísticos y Machine Learning.* Cataluña: [Trabajo Fin de Máster para optar al Título de Máster en Seguridad de las Tecnologías de la Información y de las Comunicaciones]. Universitat Oberta de Catalunya. Obtenido de http://openaccess.uoc.edu/webapps/o2/bitstream/10609/89547/6/jaruizrTFM0119memoria.pdf

Sánchez, A. (4 de mayo de 2020). *Comparativa de antivirus según protección antimalware (Q1 2020).* Obtenido de protegermipc.net: https://protegermipc.net/2020/05/04/comparativa-de-antivirus-segun-proteccion-antimalware-marzo-2020/

Stiawan, D., Susanto, Arifin, M., Idris, M., & Budiarto, R. (2020). IoT Botnet Malware Classification Using Weka Tool and Scikit-learn Machine Learning. *7th International Conference on Electrical Engineering, Computer Sciences and Informatics (EECSI)*, 15-20. Obtenido de https://ieeexplore.ieee.org/abstract/document/9251304

Trigo, S., Castellote, M., Podestá, A., Ruiz, G., Lamperti, S., & Constanzo, B. (2017). Ransomware: seguridad, investigación y tareas forenses. *Simposio Argentino de Informática y Derecho (SID)-JAIIO 46*, (págs. 1-15). Córdoba. Obtenido de http://redi.ufasta.edu.ar:8080/xmlui/bitstream/handle/123456789/1595/JAIIO%20SID%202017-2936-Ransomware-CR.pdf?sequence=1

Uchnár, M., & Feciľak, P. (2019). Behavioral malware analysis algorithm comparison. *SAMI 2019 • IEEE 17th World Symposium on Applied Machine Intelligence and Informatics*, 397-400. Obtenido de https://ieeexplore.ieee.org/document/8782717

Velasco, J. (2013). *Uso de técnicas de Web Mining: aplicación empírica en el sector de la administración pública.* Madrid, España: [Trabajo Fin de Máster en Minería de Datos e Inteligencia de Negocios]. Universidad Complutense de Madrid. Facultad de Estudios Estadísticos. Obtenido de https://eprints.ucm.es/id/eprint/25812/1/Trabajo%20Fin%20Master%20Jorge%20Velasco%20(1).pdf

Vivanco-Toala, D., Bolaños-Burgos, F., & Angulo-Murillo, N. (julio-diciembre de 2020). Estudio exploratorio de las estrategias para la protección a las redes empresariales de las

infecciones ransomware. *Revista Científica Multidisciplinaria Arbitrada YACHASUN*, *4*(7), 71-87. Obtenido de https://editorialibkn.com/index.php/Yachasun/article/view/38/96

Wigmore, I. (2021). *Internet de las cosas (IoT).* Obtenido de searchdatacenter.techtarget.com: https://searchdatacenter.techtarget.com/es/definicion/Internet-de-las-cosas-IoT

Zufiaurre, G. (2019). *Detección de malware mediante aprendizaje profundo.* Vitoria: [Trabajo Fin de Grado]. Universidad del Pais Vasco. Ingeniería en Tecnología de Telecomunicación. Obtenido de https://addi.ehu.eus/bitstream/handle/10810/36853/TFG_GLORIA_ZUFIAURRE_SOTO.pdf?sequence=1&isAllowed=y

Anexos

Anexo A. Fuentes sobre las características intrínsecas propias del malware ransomware.

Nombre	Decodificador	Info 1	Info 2	Pantallazos del Ransom
Cerber		https://blog.malwarebytes.org/threat-analysis/2016/03/cerber-ransomware-new-but-mature/	https://community.rsa.com/community/products/netwitness/blog/2016/11/04/the-evolution-of-cerber-v410	https://www.google.de/search?tbm=isch&q=Ransomware+Cerber
CryptoLocker	https://www.fireeye.com/blog/executive-perspective/2014/08/your-locker-of-information-for-cryptolocker-decryption.html	https://reaqta.com/2016/04/uncovering-ransomware-distribution-operation-part-2/		https://www.google.de/search?tbm=isch&q=Ransomware+CryptoLocker
Cryptowall		https://blogs.technet.microsoft.com/mmpc/2015/01/13/crowti-update-cryptowall-3-0/	https://www.virustotal.com/en/file/45317968759d3e37282ceb75149f627d648534c5b4685f6da3966d8f6fca662d/analysis/	https://www.google.de/search?tbm=isch&q=Ransomware+CryptoWall%203
BadRabbit	https://github.com/smartinm/diskcryptor/blob/master/boot/vc2008_src/asm/stage1.asm#L25	https://blog.malwarebytes.com/threat-analysis/2017/10/badrabbit-closer-look-new-version-petyanotpetya/	https://blog.malwarebytes.com/cybercrime/2017/10/badrabbit-ransomware-strikes-eastern-europe/	https://www.google.de/search?tbm=isch&q=Ransomware+BadRabbit

Nota: adaptado de Bazante (2019).

Anexo A. Fuentes sobre las características intrínsecas propias del malware ransomware *(continuación)*.

Nombre	Decodificador	Info 1	Info 2	Pantallazos del Ransom
Locky		http://www.bleepingcomputer.com/news/security/new-locky-version-adds-the-zepto-extension-to-encrypted-files/		https://www.google.de/search?tbm=isch&q=Ransomware+Locky
Petrwrap		https://securelist.com/blog/research/77762/petrwrap-the-new-petya-based-ransomware-used-in-targeted-attacks/		https://www.google.de/search?tbm=isch&q=Ransomware+Petrwrap
Petya	http://www.thewindowsclub.com/petya-ransomware-decrypt-tool-password-generator	https://blog.malwarebytes.org/threat-analysis/2016/04/petya-ransomware/	https://www.bleepingcomputer.com/news/security/petya-ransomware-returns-with-goldeneye-version-continuing-james-bond-theme/	https://www.google.de/search?tbm=isch&q=Ransomware+Petya

Nota: adaptado de Bazante (2019).

Anexo A. Fuentes sobre las características intrínsecas propias del malware ransomware *(Continuación).*

Nombre	Decodificador	Info 1	Info 2	Pantallazos del Ransom
Radamant	https://decrypter.emsisoft.com/radamant	http://www.bleepingcomputer.com/news/security/new-radamant-ransomware-kit-adds-rdm-extension-to-encrypted-files/	http://www.nyxbone.com/malware/radamant.html	https://www.google.de/search?tbm=isch&q=Ransomware+Radamant
Satana		https://blog.malwarebytes.com/threat-analysis/2016/06/satana-ransomware/	https://blog.kaspersky.com/satana-ransomware/12558/	https://www.google.de/search?tbm=isch&q=Ransomware+Satana
TeslaCrypt	http://www.bleepingcomputer.com/forums/t/576600/tesladecoder-released-to-decrypt-exx-ezz-ecc-files-encrypted-by-teslacrypt/	https://www.endgame.com/blog/your-package-has-been-successfully-encrypted-teslacrypt-41a-and-malware-attack-chain	https://blog.kaspersky.com/raknidecryptor-vs-teslacrypt/12169/	https://www.google.de/search?tbm=isch&q=Ransomware+TeslaCrypt
WannaCry		https://docs.google.com/spreadsheets/d/1XNCCiiwpIfW8y0mzTUdLLVzoW6x64hkHJ29hcQW5deQ/pubhtml#	https://twitter.com/struppigel/status/846241982347427840	https://www.google.de/search?tbm=isch&q=Ransomware+WannaCry

Nota: adaptado de Bazante (2019).

Anexo A. Fuentes sobre las características intrínsecas propias del malware ransomware *(Continuación).*

Nombre	Decodificador	Info 1	Info 2	Pantallazos del Ransom
Dopplepaymer		https://www.pcrisk.es/guias-de-desinfeccion/9454-doppelpaymer-ransomware		https://www.google.de/search?q=Ransomware+Dopplepaymer&tbm=isch&ved=2ahUKEwjX8L611oLwAhU1QTABHUEtBxIQ2-cCegQIABAA&oq=Ransomware+Dopplepaymer&gs_lcp=CgNpbWcQA1CGoqYBWIaipgFgoqumAWgAcAB4AIABpwKIAacCkgEDMi0xmAEAoAEBqgELZ3dzLXdpei1pbWfAAQE&sclient=img&ei=6XZ5YJfYC7WCwbkPwdqckAE
Maze		https://www.pcrisk.es/guias-de-desinfeccion/9512-maze-2019-ransomware	https://blog.malwarebytes.com/threat-spotlight/2020/05/maze-the-ransomware-that-introduced-an-extra-twist/	https://www.google.de/search?q=Ransomware+Maze&tbm=isch&ved=2ahUKEwiE7ezJ4ILwAhVqcTABHdn-AyIQ2-cCegQIABAA&oq=Ransomware+Maze&gs_lcp=CgNpbWcQAzICCAAyBggAEAgQHjIGCAAQCBAeMgYIABAIEB4yBggAEAgQHjoECAAQHlCgl0NY49JDYMnuQ2gAcAB4AIABoAKIAboYkgEFMC44LjiYAQCgAQGqAQtnd3Mtd2l6LWltZ8ABAQ&sclient=img&ei=kIF5YITlDuriwbkP2f2PkAI
Revil/Sodinokibi		https://www.pcrisk.es/guias-de-desinfeccion/9153-sodinokibi-ransomware	https://blog.malwarebytes.com/cybercrime/2020/05/sodinokibi-drops-greatest-hits-collection-and-crime-is-the-secret-ingredient/	https://www.google.de/search?q=Ransomware+Revil%2FSodinokibi&tbm=isch&ved=2ahUKEwiQhdDc5ILwAhX8cTABHYdXBrwQ2-cCegQIABAA&oq=Ransomware+Revil%2FSodinokibi&gs_lcp=CgNpbWcQA1DroChY66AoYJ6oKGgAcAB4AIABeIgBeJIBAzAuMZgBAKABAaoBC2d3cy13aXotaW1nwAEB&sclient=img&ei=6YV5YNDsFfzjwbkPh6-Z4As

Nota: adaptado de Bazante (2019).

Anexo A. Fuentes sobre las características intrínsecas propias del malware ransomware *(Continuación).*

Nombre	Decodificador	Info 1	Info 2	Pantallazos del Ransom
Netwalker		https://www.pcrisk.es/guias-de-desinfeccion/9587-mailto-ransomware		https://www.google.de/search?q=Ransomware+Netwalker&tbm=isch&ved=2ahUKEwiklJ2Y54LwAhUFbjABHVbXDdkQ2-cCegQIABAA&oq=Ransomware+Netwalker&gs_lcp=CgNpbWcQAzIGCAAQCBAeMgYIABAIEB5Qm4UXWJuFF2DHjBdoAHAAeACAAboBiAG6AZIBAzAuMZgBAKABAaoBC2d3cy13aXotaW1nwAEB&sclient=img&ei=f4h5YOTYDYXcwbkP1q63yA0
Egregor		https://www.pcrisk.es/guias-de-desinfeccion/10073-egregor-ransomware	https://www.trendmicro.com/en_us/research/20/l/egregor-ransomware-launches-string-of-high-profile-attacks-to-en.html	https://www.google.de/search?q=Ransomware+Egregor&tbm=isch&ved=2ahUKEwia1Iu91oLwAhVmeTABHQbdAmoQ2-cCegQIABAA&oq=Ransomware+Egregor&gs_lcp=CgNpbWcQAzICCAA6BAgAEB5Q1-DeAli88N4CYPyC3wJoAHAAeACAAeYBiAHQA5IBBTAuMi4xmAEAoAEBqgELZ3dzLXdpei1pbWfAAQE&sclient=img&ei=-XZ5YNrzB-bywbkPhrqL0AY

Nota: adaptado de Bazante (2019).