Institución Universitaria

Esquema metodológico apoyado en una herramienta (software) para la detección y prevención de Crypto Ransomware en una estación de trabajo

Andrés Felipe Osorio Sierra

Instituto Tecnológico Metropolitano

Facultad de Ingenierías

Medellín, Colombia

2018

Esquema metodológico apoyado en una herramienta (software) para la detección y prevención de Crypto Ransomware en una estación de trabajo

Andres Felipe Osorio Sierra

Tesis o Trabajo de investigación presentada(o) como requisito parcial para optar al título de:

Magister en Seguridad Informática

Director (a):

Ph.D. Carlos Andres Mera Banguero

Director (a):

M.Sc. Milton Javier Mateus Hernandez

Línea de Investigación:

Ciencias Computacionales

Instituto Tecnológico Metropolitano

Facultad de Ingenierías

Medellín, Colombia

2019

A DIOS por haberme regalado la vida.
A mi padre, porque sé que se sentirá orgulloso de verme lograr este sueño.
A mi madre, porque los esfuerzos que juntos hemos realizado.
A mi abuela, por enseñarme el amor que lleva a triunfar.
A la memoria de mis abuelos, por darme la alegría y el secreto de la felicidad.
A mi esposa, por su paciencia y su entrega incondicional.
A mi hija, para que algún día vea en mí un hombre y un ejemplo a seguir.
A mi familia por siempre permanecer unidos.
A mis profesores por todo lo que me enseñado.

Agradecimientos

Quiero agradecer, especial y enormemente, Carlos Andrés Mera, Milton Javier Mateus y Héctor Vargas por creer en mí durante todo el proceso de desarrollo de este trabajo de investigación. Por su apoyo, aportes, recomendaciones y la paciencia que tuvieron conmigo, por las discusiones generadas que hicieron posible la construcción de este trabajo. También agradezco a la Institución Universitaria ITM y el grupo de profesores por brindarme todo su conocimiento y disposición por lograr uno de mis sueños y que hoy hace posible se logren.

Resumen

En los últimos años, los *malware* tipo *ransomware* han demostrado ser una amenaza de seguridad para las empresas y personas, esto se debe a que los métodos de detección y prevención son insuficientes y las variantes de *ransomware* actúan de manera diferente debido a los diferentes vectores de ataque que utiliza para comprometer un equipo, por lo que comprender el comportamiento y funcionamiento del gran numero variables es complicado, en esta investigación se tomaron 22 muestras representativas y fueron probadas en un ambiente controlado con el fin entender el proceso del ciclo de vida del *ransomware*. La metodología propuesta para la detección y prevención de *ransomware* se elaboró con las bases científicas existentes de los diferentes métodos de detección y prevención *ransomware* y apoyado mediante una herramienta o software desarrollada en el lenguaje de programación *Python* y el marco lógico escalable llamado (*Malice)* minimizando el impacto negativo que tiene el *ransomware* en las empresas u hogares.

En la investigación se presenta la formulación de un esquema metodológico basado en la detección y prevención de *malware* tipo crypto *ransomware*, se desarrolló mediante la búsqueda de los métodos existentes determinando la efectividad a la hora de detectar y prevenir un *ransomware*. Se inició con la selección y caracterización de los criterios y variables más comunes del *ransomware* mediante el análisis dinámico de las variantes de *ransomware* que fueron usadas para conocer el origen y evolución que ha tenido dicho *malware.* Una vez entendiendo el comportamiento del *ransomware* se agruparon las acciones que combaten los patrones de comportamiento de cada variante de *ransomware,* y a partir de ahí se empezó con la conceptualización de los diferentes métodos de detección y prevención de *ransomware*, logrando el diseño del esquema metodológico que reunió todos los métodos o acciones para la detección y prevención de *ransomware* en una estación de trabajo. Por último se inició el desarrollo de un software basado en alguno de los métodos propuestos del esquema metodológico, además valoramos la efectividad del método detección y prevención con respecto de los patrones de comportamiento establecidos. Con la solución propuesta se generaron nuevos mecanismos para la prevención y detección de los nuevos tipos de *crypto ransomware*.

Palabras clave: *malware, ransomware, crypto ransomware*, *carga útil, esquema, detección de ransomware, metodología de detección*

Abstract

In recent years, *ransomware* type malware has proven to be a security threat to businesses and individuals, this is because the detection and prevention methods are insufficient and the variants of *ransomware* act differently due to different attack vectors used to compromise a computer, so understanding the behavior and operation of the large number of variables is complicated, in this research were taken 22 representative samples and were tested in a controlled environment in order to understand the process of the life cycle of *ransomware*. The methodology proposed for the detection and prevention of *ransomware* was elaborated with the existing scientific bases of the different methods of detection and prevention *ransomware* and supported by a tool or software developed in the programming language Python and the scalable logical framework called (Malice) minimizing the negative impact that *ransomware* has in companies or homes.

The research presents the formulation of a methodological scheme based on the detection and prevention of crypto *ransomware* type badware, was developed by searching for existing methods determining the effectiveness in detecting and preventing *ransomware*. It began with the selection and characterization of the most common criteria and variables of ransomware through the dynamic analysis of the variants of *ransomware* that were used to know the origin and evolution of this *badware*. Once the behavior of *ransomware* was understood, the actions that combat the behavior patterns of each variant of *ransomware* were grouped together, and from there the conceptualization of the different methods of detection and prevention of *ransomware* began, achieving the design of the methodological scheme that brought together all the methods or actions for the detection and prevention of *ransomware* on a workstation. Finally, the development of a software based on some of the methods proposed in the methodological scheme was started. We also assessed the effectiveness of the detection and prevention method with respect to the established behavior patterns. The proposed solution generated new mechanisms for the prevention and detection of new types of crypto *ransomware*.

Keywords *ransomware*, *badware*, payload, bitcoins, schema ,ransomware detection, detection methodology

Contenido

Lista de figuras

Lista de tablas

Lista de Símbolos y abreviaturas

Abreviatura	**Término**
CR	Crypto *Ransomware*
NCR	*Ransomware* No Criptografico
C&C	Centro de control y comando
DGA	Algoritmo de generación de dominios
IoC	Indicador de compromiso
AV	Antivirus
IDS	Sistema de Detección de Intrusos
IPS	Sistema de Prevención de Intrusos
NIST	National Institute of Standards and Technology
DLP	Prevención de Pérdida de Datos
NAC	Control de Acceso en Red
IoT	Internet de las cosas
FTP	Protocolo de Transferencia de archivos
DEP	Prevención de Ejecución de Datos
RDP	Protocolo de Escritorio Remoto
SSL	Capa de Sockets Seguros
TLS	Seguridad en Capa de Transporte
UVA	Análisis de Comportamiento de Usuarios
SIEM	Gestión de Eventos de Información de Seguridad
SRP	Política de Restricción de Software
FSRM	Windows File Services Resource Manager
CVSS	Sistema de puntuación de vulnerabilidad común

Introducción

El auge tecnológico, el desarrollo y la masificación del uso de Internet han facilitado la forma cómo compartimos y accedemos a la información. Estos desarrollos han mejorado, en muchos aspectos, el estilo de vida de las personas y la productividad de las empresas, sin embargo, también han traído diferentes amenazas (Melo, 2011), entre ellas los *malware*. Un *malware* es un software malicioso que está diseñado para recopilar información sensible, para acceder a los sistemas de archivos privados de un usuario o para interrumpir el funcionamiento normal de una computadora (Eset, 2016).

Un tipo especial de *malware*, llamado *ransomware*, es un software malicioso que ha sido creado con la intención de secuestrar los datos de un usuario, para luego pedir un rescate por estos (Kevin Savage, Peter Coogan, & Hon Lau, 2015). Dadas las implicaciones de los daños y las repercusiones económicas que puede traer consigo el secuestro de información, la ciberseguridad se ha convertido en un área de interés que atrae no solo al personal de seguridad de las empresas, sino también a investigadores y desarrolladores para participar en la búsqueda de contramedidas y mitigar el impacto de los diferentes tipos de *malware* existentes.

Uno de los problemas más complejos cuando se trata de ataques por *malware* es que los atacantes desarrollan mutaciones de sus versiones iniciales que dificultan la mitigación de los ataques. Para el caso específico de *ransomware*, el aumento de familias y variantes en el último año es exponencial (J. A. Gómez-Hernández, Pedro García-Teodoro, & L. Álvarez-González, 2018). (J. A. Gómez-Hernández et al., 2018). Por ejemplo, en el caso de WannaCry, un *ransomware* que apareció en mayo de 2017 y que ha impactado considerablemente a personas y empresas en el último año, se ha estimado que las diferentes variantes de *ransomware* han generado pérdidas de alrededor de los 200 millones de euros (IBM, 2016).

En general, para desarrollar un agente que evite la infección de algún *malware* determinado, se hace necesario entender no sólo su estructura, sino también su comportamiento. Una estrategia comúnmente utilizada para esto es el análisis dinámico (Gonzalez & Hayajneh, 2017). Este tipo de análisis consiste en obtener información sobre los archivos que el *malware* crea en el sistema

operativo, las conexiones de red que éste establece y las modificaciones que realiza en registro del sistema operativo, entre otras. En la actualidad existen diferentes herramientas que permiten a una persona hacer el análisis dinámico de un *malware*, entre ellas, Regshot, Psviewer y Wireshark. Este tipo de herramientas se usan para el análisis de *malware* del tipo *ransomware*, sin embargo, su uso debe hacerse en un entorno aislado del sistema operativo, por ejemplo, en una máquina virtual que corra una copia del sistema operativo o mediante un *sandbox* que permita el aislamiento de los procesos que el *ransomware* realiza para secuestrar los datos.

Dado el impacto negativo del *malware* de tipo *ransomware* y debido a que aún los esquemas de detección para este tipo de *malware* son independientes, desagrupados y no hay una integración de cada uno de estos, en este trabajo de grado se presenta una metodología que, partiendo del uso de un análisis dinámico de algunas de las variantes seleccionadas de *ransomware*, comprendiendo el comportamiento, técnicas utilizadas y fases , generar un esquema metodológico con los controles de prevención y detección para la neutralización del *ransomware*. Luego se procede a la creación de una herramienta de detección y prevención en base a alguno de los controles descritos y puedan adecuarse en una estación de trabajo y mitigar el impacto generado por el *ransomware.*

1.1 Definición del Problema

Actualmente existen soluciones de seguridad que permiten evitar los incidentes y eventos que se generan en las organizaciones y empresas a causa del *malware*, aun así, los sistemas informáticos siguen siendo susceptibles a algunos tipos de amenazas como los *zero-day* o *malware* diseñados para afectar a un sector económico o enfocados a infraestructuras críticas.

Un aspecto importante es que las soluciones de seguridad existentes están enfocadas a estaciones del usuario final. Entre dichas soluciones están los antivirus, HIPS, firewall y a nivel de red mediante proxys, firewalls, IPS/IDS, UTM y sistemas antispam, entre otras. También hay algunas soluciones que surgen a partir de las nuevas tecnologías en la nube como *sandbox*, CDN, técnicas de aprendizaje de máquina integradas a los antivirus, detección de botnets, antiddos, control de redes de centro de comandos e indicadores de compromiso con lo cual logran adicionar proteger la red, equipos y usuarios (Gaviria Pablo, 2016).

Aun así, la constante aparición de nuevos tipos de *malware* y la forma como mejoran las técnicas de ocultamiento de *malware* mediante la ofuscación que permite evadir la detección por firmas, ha dificultado la prevención y detección de las nuevas variantes de *malware* entre ellas el *ransomware*. Esto evidencia la necesidad de desarrollar herramientas y métodos para ayudar a detectar y prevenir la infección de este tipo de *malware*.

Con base en lo anterior, este trabajo propone un esquema metodológico para la detección y prevención de *ransomware* durante las diferentes fases o etapas de su ciclo de vida. La metodología propuesta apunta a ayudar a los analistas de seguridad a ejecutar acciones y ejercer controles para aquellos archivos ejecutables que presenten algún comportamiento de *malware* tipo *ransomware,* los controles y métodos serán enumerados.

1.2 Objetivos

1.2.1 Objetivo General

Formular un esquema metodológico apoyado en una herramienta (software) para la detección y prevención de *Crypto Ransomware* en una estación de trabajo.

1.2.2 Objetivo Específicos

- Caracterizar los criterios y variables del *Crypto Ransomware* para elegir los que sean pertinentes y puedan ser usadas en el esquema metodológico en la detección y prevención.
- Diseñar un esquema metodológico que reúna las acciones para la detección y prevención de *ransomware* una estación de trabajo.
- Desarrollar una herramienta (software) basado en el diseño del esquema metodológico propuesto para la detección y prevención de Crypto Ransomware.
- Validar el esquema metodológico propuesto a partir de pruebas con muestras seleccionadas del *Crypto-Ransomware* para medir sus fortalezas y debilidades.

1.3 Alcances

El alcance del trabajo de grado se enmarca en la creación de un esquema mediante los métodos de detección y prevención de *ransomware* existentes, analizando los conceptos teóricos y prácticos desarrollados en el campo de la seguridad informática, dichos métodos serán los mecanismos y controles que permitan ser implementados en una estación de trabajo y ser adaptados para formulación del esquema metodológico propuesto.

El esquema metodológico propuesto analiza los estudios previos publicados por autores que explican los diferentes métodos para la detección y prevención del *ransomware* en las fases de ciclo de vida de un ataque de *malware,* especialmente el comportamiento de este. La necesidad de profundizar en el esquema de detección y prevención se debe a que no han sido estudiados y analizados los mecanismos de detección y prevención para *ransomware* de manera estructurada bajo un esquema metodológico que agrupe todos los métodos de detección y prevención de

ransomware. Esto se debe a que estos son métodos que proponen los autores funcionan de manera independiente, no tienen integración y no hay un orden categorizado que enmarque el conjunto de métodos teórico prácticos que puedan servir para la elaboración de un esquema de detección y prevención.

La validación del esquema propuesto se hará con base en un conjunto de muestras de *malware* tipo *ransomware* previamente seleccionadas. La evaluación se realizará respecto a la capacidad que tiene la herramienta de software desarrollada para detectar y prevenir, en una estación de trabajo, mediante la elección de las variables de *ransomware*.

1.4 Estructura del documento

Esta tesis está dividida en 2 partes y agrupados por capítulos 4 Capítulos.

En la Parte I se presenta el problema objeto de estudio, los objetivos y el alcance de esta tesis.

En la Parte II se presentan todos los capítulos o los objetivos de este trabajo de grado.

- En el Capítulo 2 se introducen los conceptos fundamentales e ideas del *ransomware*, además se realizará una selección, caracterización y categorización de las variables de *ransomware* mediante la clasificación de 8 variables seleccionadas, esto debido a la gran variedad de *ransomware*. También se realiza la descripción de los comportamientos y patrones que tienen cada una de las variantes seleccionadas, identificando los vectores de ataque, funcionalidades y comportamientos que tienen, esto para dar inicio a un análisis de cada una de las variantes y comprender cuales son los métodos de ataque y distribución para entender mejor su comportamiento.

- En el Capítulo 3 se presenta el diseño y el esquema metodológico mediante la agrupación de las acciones de prevención y detección de *ransomware*, además se muestra el diagrama y el diseño del esquema metodológico para la detección y prevención de este en una estación de trabajo y se realizará una labor experimental usando casos de prueba con muestras de *ransomware* provenientes de repositorios de *malware* teniendo en cuenta los métodos de prevención y detección de *ransomware*.

- En el Capítulo 4 se presenta el desarrollo de la herramienta para la prevención y detección mediante la clasificación de los diferentes framework, se eligió a Malice para la implementación de los scripts mediante Python, además este framework permite llevar a cabo las pruebas en una entorno controlado.

- En el Capítulo 5 se realiza la validación del esquema metodológico mediante la comparación de los diferentes métodos de prevención y detección frente a cada una de las variables determinando el grado de efectividad. Se utilizará la estadística descriptiva para la recopilación y recuento de la información, así como también se procederá a la sistematización y análisis de la información levantada, con el propósito de correlacionar y validar si el software es efectivo partiendo de las tablas generadas.

2. Marco Teórico y Estado del Arte

En los siguientes capítulos se presentan el marco teórico y el estado del arte con los conceptos relacionados con el *malware* tipo *ransomware*. Dichos conceptos permiten determinar cuáles son los elementos importantes para definir un esquema de detección y prevención de este tipo de *malware*. Adicionalmente, se describen otros trabajos que existen en la literatura y que están relacionados con el objeto de estudio de este trabajo de grado.

2.1 Definición de *Ransomware*

El *ransomware* es un tipo de *malware* de extorsión mediante el cual los atacantes secuestran y cifran los archivos de la computadora de la víctima (TrendMicro, 2011). En este sentido, el propósito de un *ransomware* es exigir un pago monetario, generalmente a través de bitcoins, para el rescate de los archivos. Varían según el tipo de *ransomware* dependiendo del cobro de la extorsión donde incluyen opciones de pago alternativas, como las tarjetas de regalo de iTunes y Amazon o bitcoins. No obstante, el pago de un rescate por los datos no garantiza que los usuarios obtengan la clave de descifrado o la herramienta de desbloqueo necesarias para recuperar el acceso a los datos de la máquina infectada (TrendMicro, 2016).

En general, el *ransomware* explota las vulnerabilidades del sistema operativo y del usuario para introducirse en la computadora y encriptar todos sus archivos. De esta manera el atacante mantiene secuestrados los archivos de la víctima hasta que esta pague el rescate de los mismos. A pesar de las muchas variantes de *ransomware*, se ha identificado que estos siguen un comportamiento similar. Primero, el *malware* busca ciertos archivos con extensiones como: .txt, .doc,. rft, .ppt, .chm, .cpp, .asm, .db, .db1, .dbx, .cgi, .dsw, .gzip, .zip, .jpg, .key, .mdb, .pgp y .pdf,

entre otras. Una vez identificados los archivos se inicia el proceso de cifrado (simétrico o asimétrico) de los mismos para limitar su acceso. A continuación, el atacante envía a la víctima una solicitud de rescate, bien sea por correo electrónico o usando una ventana emergente que exige la clave de cifrado que desbloquea los archivos congelados (Jatinder N.D. Gupta, 2008). Entre los ejemplos más conocidos de *ransomware* que afectan a las computadoras de escritorio están WannaCry, Petya, XData, Reveton, CryptoLocker, CryptoWall y TeslaCrypt. Los que afectan las plataformas móviles incluyen Simplocker y LockerPin (Salvi & Kerkar, 2017).

2.2 Tipos de *Ransomware*

Como se ha mencionado, existen diferentes variantes de *ransomware*, sin embargo, estas pueden ser agrupadas en tres categorías o tipos (Gonzalez & Hayajneh, 2017): *crypto ransomware* (CR), locker *ransomware* o también conocidos como no criptográficos (NCR) y *crypto ransomware* de llave privada.

(Bhardwaj, Subrahmanyam, Avasthi, & Sastry, 2015) y (Kevin Savage et al., 2015) plantean 2 definiciones Locker y *crypto ransomware* (CR) este último tiene la misma funcionalidad del *crypto ransomware* de llave privada debido a que usan algoritmos de llaves públicas y privadas para cifrar la información, por este motivo y para efectos de la investigación tomaremos solo 2 definiciones.

2.2.1 *Crypto Ransomware* o (CR)

Los *crypto ransomware* utilizan algoritmos criptográficos simétricos y asimétricos para cifrar los archivos, el *malware* comienza a cifrar los datos y archivos del usuario; una vez completado el cifrado, la víctima es informada de que todos sus datos están cifrados y sólo se pueden descifrar si paga el rescate. Las primeras versiones de *crypto ransomware* almacenaban la clave de descifrado en el computador. Esto permitía recuperar dicha clave mediante ingeniería inversa. La evolución de este tipo de *ransomware* se contactan con un servidor para enviar las llaves de cifrado. Algunos *crypto ransomware* son más agresivos y no solo encriptan los datos sino que también realizan otras acciones, por ejemplo, eliminar los archivos en el sistema infectado, si el pago no se realiza dentro de un plazo determinado (Gonzalez & Hayajneh, 2017).

2.2.2 Locker *Ransomware* o *Ransomware* no criptográfico (NCR)

Este tipo de *ransomware* niega el acceso al dispositivo, es decir que bloquea la interfaz de usuario del dispositivo y luego solicita a la víctima el rescate. Esta variante de *ransomware* deja a la víctima con muy pocas capacidades, por ejemplo, sólo le permite comunicarse con el atacante o realizar el pago del rescate.

A diferencia de los otros tipos de *ransomware*, los locker no utilizan ninguna clase de cifrado. En su lugar, estos restringen por completo la interacción con el sistema operativo, bien sea bloqueando la pantalla, modificando el registro de arranque maestro (MBR) o la tabla de particiones del equipo infectado. Esta característica hace que los *locker ransomware* sean considerados relativamente débiles porque el daño puede ser revertido sin pagar el rescate (Gonzalez & Hayajneh, 2017).

2.3 Evolución del *Ransomware*

La Figura 2-1, tomada de (Monika, Zavarsky, & Lindskog, 2016) muestra gráficamente cual ha sido la evolución y el incremento de variantes de *ransomware* entre los años 1989 y 2016. El incremento de las variables de *ransomware* consta de 32000 variantes nuevas y 337205 ataques generados por ransomware. (Kaspersky Lab, 2016).

Figura 2-1. Evolución del *ransomware.*

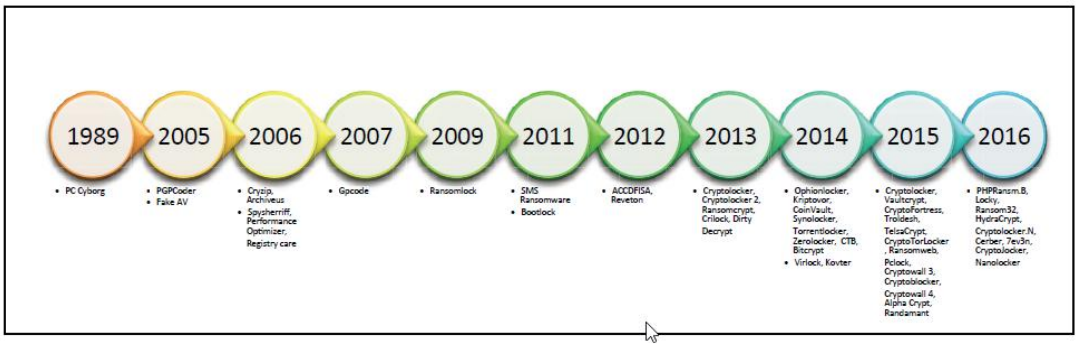


Figura tomada de (Monika et al., 2016).

Aunque los conceptos teóricos iniciales fueron propuestos por Young y Yung en el año de 1996 (Young & Yung, 1996), el primer *malware* tipo *ransomware* apareció en el año de 1989 (Brewer, 2016), (Gazet, 2010), cuando Joseph Popp creó un virus denominado AIDS, también conocido como *PC Cyborg Trojan*. Popp distribuyó dicho virus a través de 20000 diskettes que envió a investigadores del SIDA, a diferentes países del mundo. El atacante indicó a los investigadores que los diskettes contenían un programa que les ayudaría a analizar el riesgo que podía tener una persona para adquirir el virus del SIDA. Una vez instalado, el *malware* permanecía inactivo en los equipos y sólo se activaba después de que el equipo se encendiera 90 veces, punto en el cual exigía el pago de 180 dólares para permitir el acceso al equipo infectado. Dicho pago debía realizarse a una oficina de cobro en la ciudad de Panamá.

Más adelante, en el año 2005, apareció un virus denominado PGPCoder. Este *malware* infecta a los equipos a través de un archivo, llamado anketa.doc, que se adjuntaba a los correos electrónicos. El archivo contenía una macro maliciosa que cifraba todos los archivos con extensiones .doc, .xls, .pdf, .ppt, entre otras. Una vez encriptados los archivos, PGPCoder autodestruía el archivo infectado y creaba un nuevo archivo llamado readme.txt que proporcionaba la información sobre cómo contactarse con el atacante para recuperar los archivos cifrados (Denis Nazarov, Olga Emelyanova, 2006).

Luego, en el año 2006, apareció Archiveus, el primer *ransomware* que utiliza un cifrado asimétrico, junto con el algoritmo RSA para bloquear el acceso a los archivos de la máquina infectada. Para

recuperar sus archivos, las víctimas tenían que comprar una contraseña de descifrado en ciertos sitios web específicos. Una particularidad de Archiveus es que sólo cifraba los archivos en la carpeta "Mis Documentos" en los equipos basados en Windows.

A partir de 2008, los *ransomware* se convirtieron en un problema puesto que empezaron a engañar y a persuadir a los usuarios para que descargaran software falso, donde instalaba una copia del virus en la máquina. La cuestión con los falsos instaladores es que estos se veían y actuaban casi de la misma manera a como lo hacían sus contrapartes legítimas. No obstante, una vez se instalaba el virus, este solicitaba hasta 100 dólares para solucionar el problema en la máquina infectada (Smith, 2016).

Para el año 2009 apareció el Ransomlock, el cual fue uno de los primeros *ransomware* del tipo locker. Esta variante del *malware* tenía como objetivo bloquear el acceso al computador infectado hasta que se comprara un programa determinado o, se enviara un SMS (Servicio de Mensajes Cortos) a un número determinado que se proporcionaba en la pantalla de la máquina infectada.

Después, en el año 2012 surgió Reveton, un *ransomware* basado en el troyano Citadel (Krunal & Viral, 2017), que cifra los archivos del usuario y muestra un mensaje en la pantalla indicando que el computador ha sido usado para actividades ilícitas, tales como descargas de software pirata o pornografía infantil. Con este mensaje, el virus hace creer al usuario que el remitente del mismo es una agencia del estado. Una vez el usuario accedía a los enlaces que se proporcionaban en el mensaje, este llevaba a sitios web comprometidos en los que se exigía el pago a través de tarjetas de pago (Kharraz, Robertson, Balzarotti, Bilge, & Kirda, 2015).

Luego, con la creación del Bitcoin, en 2009, se abrió un método anónimo de extorsión que antes no estaba disponible para los atacantes. Esto llevó a que en el 2013 apareciera CryptoLocker, un *ransomware* que se propagó rápidamente a través de sitios web comprometidos y archivos que llegaban mediante adjuntos de correos electrónicos maliciosos. CryptoLocker utilizó el algoritmo de cifrado AES-256 cifrando los archivos y documentos de equipos infectados mediante una central de comando donde realizaba ataques distribuidos a través de la red de Bots Zeus, además, distribuía las claves de descifrado usando la red TOR, cabe resaltar que fue el primero en usar TOR y cobrar en Bitcoins.

Para el año 2014, surgieron las primeras medidas contra los *malware* tipo *ransomware. Esto* debido al incremento de ataques y a la aparición de variantes como CryptoWall, que se distribuían de forma masiva y generaron ingresos estimados de 325 millones de dólares para los ciberdelincuentes. También apareció CryptoDefense, que usaba encriptación RSA de 2048 bits pero dejaba la clave de desencriptación en texto plano en la computadora. CryptoWall empleaba kits de explotación y era más difícil de erradicar porque podía copiarse a sí mismo en las claves del registro y en las carpetas de inicio.

Para los años 2015 y 2016 aparecieron los primeros ataques a dispositivos móviles con el lanzamiento de LockerPin, un *ransomware* que cambia el PIN de acceso de los teléfonos con el sistema operativo Android. Específicamente, este *ransomware* exigía una suma de 500 dólares a las víctimas para desbloquear el dispositivo.

En estos mismos años también aparecieron los *ransomware* diseñados para los usuarios de Linux. Encoder fue el primer *ransomware* programado para atacar los sistemas de alojamiento web basados en Linux, bloqueando los directorios web y encriptando en el contenido aplicaciones como Magento y cPanel. Durante 2006 un *ransomware* llamado Chimera tenía la particularidad de que no sólo encriptaba archivos, sino que también amenazaba con publicarlos en línea si no se pagaba el rescate, esta práctica es conocida como doxing.

En el año 2017 se registró el primer ciberataque mundial con *ransomware* mediante Wannacry, el cual infectó a más de 250.000 dispositivos utilizando técnicas de la herramienta de hacking EternalBlue filtrada por agentes de la NSA y una vulnerabilidad del protocolo SMB de Windows (Kharraz, Robertson, Balzarotti, Bilge, & Kirda, 2015).

2.4 Esquema de Funcionamiento de un *Ransomware*

Para el análisis del comportamiento de un *ransomware* es importante conocer las nuevas técnicas usadas por los ciberdelincuentes que cada vez mejoran las variantes de *ransomware*, por ende se necesita comprender su funcionamiento y entender las nuevas formas de las variaciones y mutaciones de este, además ayudan en la detección temprana de *ransomware* para proteger los archivos. Para realizar el análisis de comportamiento de un *ransomware* se usa cualquiera de las siguientes fases: Análisis dinámico, análisis estático, análisis mediante comportamiento e ingeniería inversa. Para efectos de esta investigación usaremos la fase de análisis dinámico (Gaviria Pablo, 2016)

Durante el análisis dinámico del *ransomware* se observa que tienen una estructura y comportamiento similares a otros tipos *malware*, como troyanos o gusanos que contienen técnicas de ofuscación, similitud en los *payload*, autoreplicación, técnicas de ocultamiento, persistencia, creación de registros, creación de carpetas, generar tráfico en red y cifrado. A continuación, se describen los componentes de un *ransomware* de manera más detallada.

Inicialmente el *malware* requiere un *payload*, que es el código que se caracteriza por ejecutar el daño en un sistema operativo. Es decir, el payload es el componente principal y es el que se trata de ocultar para que no pueda ser detectado por las firmas de los antivirus (Nieuwenhuizen, 2017). Dependiendo del tipo de *malware*, las funciones del payload pueden ser, entre otras, robar los nombres de usuario y contraseñas almacenadas en la máquina, descargar otros *malware* que potencialicen sus capacidades y mostrar mensajes de alertas de que el sistema puede estar infectado.

Para ocultar el payload a los antivirus se requiere del uso de técnicas de ocultamiento y ofuscación que permitan al *malware* ejecutarse de forma sigilosa y rápida. Para el caso de *ransomware* se suelen utilizar técnicas de ofuscamiento mediante el uso de un empacador que comprime y altera el código original del *malware* (Philip OKane ; Sakir Sezer;Kieran McLaughlin, 2011). De esta forma se engaña al antivirus, el cual “ve” un código muy diferente al código original del *malware*.

Adicionalmente, algunos *ransomware* usan mecanismos de protección y blindaje para evitar que otros programas identifiquen las acciones y el comportamiento que este va realizando durante su ejecución. De esta forma el *malware* dificulta el trabajo de los analistas de seguridad o de los forenses digitales.

Un elemento importante de los *ransomware* es la persistencia. Esta permite al *malware* seguir funcionando cuando se reinicie el sistema operativo. Para ello, los *malware* modifican los registros del sistema operativo o de procesamiento, habilitan o deshabilitan ciertos servicios, agregan tareas programadas y crean carpetas o ejecutables de inicio del sistema operativo, entre otros.

Por otro lado, algunos tienen la capacidad de desactivar o eliminar la funcionalidad de restauración del sistema, incluso algunos eliminan las copias shadow de Windows para evitar que los datos cifrados se restauren a una versión anterior de Windows (Hosmer, Bartolomie, & Pelli, 2016).

Otros tipo de *ransomware*, como Petya, modifican el sector de arranque del sistema operativo para limitar o evitar el acceso al mismo. Estos *ransomware* son conocidos como *Boot ransomware.*

Cuando se ejecuta el ransomware, este debe mapear el entorno del sistema para iniciar su configuración e instalación. Esto se hace normalmente realizando un mapeo del sistema operativo para determinar si se está ejecutando en un equipo real o en un entorno de virtualización.

Una de las tareas importantes es la creación de archivos en el registro de Windows que permiten al *ransomware* pueda funcionar de manera persistente, esté requiere alterar y modificar el registro de Windows donde se almacenarán las cadenas de registro y para cuando se reinicie el ordenador y mantenga la posibilidad de auto ejecutarse para poder seguir teniendo control del equipo. Los archivos del registro, comúnmente utilizados por los *ransomware* son:

HKCU\Software\Microsoft\Windows\CurrentVersion\RunOnce
HKLM\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer\Run
HKCU\Software\Microsoft\Windows NT\CurrentVersion\Winlogon\Shell
HKLM\Software\Microsoft\Windows NT\CurrentVersion\Winlogon\Shell
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\Winlogon\Userinit

Al igual que el elemento anterior, la creación de carpetas está incluida en cada *ransomware* para crear su propia estructura de carpetas, las cuales tienen el propósito de almacenar los ejecutables necesarios para iniciar su actividad, algunas carpetas utilizadas el *ransomware* es C:\DocumentsandSettings \user\Start Menu\Programs\Startup.

La inyección de procesos es una técnica que se utiliza para engañar al sistema operativo, que incluye la inyección de procesos legítimos que se ejecutan por lo general en el directorio %AppData% y mediante ejecutables falsos con el mismo nombre de un ejecutable de Windows alteran el funcionamiento del sistema. Para el funcionamiento de inyección de procesos se generan a través de dos técnicas: la inyección DLL o inyección directa (Caivano, Canfora, Cocomazzi, Pirozzi, & Visaggio, 2017).

Cuando el *ransomware* habilita o deshabilita los servicios del sistema lo hace para dañar y persuadir al sistema operativo. Es decir, el ransomware busca los servicios y procesos que pueden detectarlo, para posteriormente habilitar o deshabilitarlos. Normalmente para modificar un servicio, los *ransomware* generan una entrada de registro en las siguientes rutas:

HKLM\SYSTEM\CurrentControlSet\services
HKLM\Software\Microsoft\Windows\CurrentVersion\Run\Services
HKLM\Software\Microsoft\Windows\CurrentVersion\Run\Services\Once

Por otra parte, para mejorar las capacidades del *ransomware* es importante la elevación de privilegios**,** esto para la ejecución de actividades maliciosas adicionales en el sistema. Una técnica de elevación de privilegios comúnmente usada es SeDebugPrivilege, que permite al *malware* acceder a nivel de sistema estableciendo los derechos de usuario mediante un token de acceso. Un token de acceso en un sistema Windows, es un objeto que contiene descriptores de seguridad de un proceso (Caivano et al., 2017).

Durante la etapa final de comportamiento de un *ransomware* usa el cifrado**,** que es el proceso mediante el cual el *ransomware* genera los algoritmos de cifrado y la generación de las llaves privadas o públicas para realizar cifrar los archivos (Monika et al., 2016).

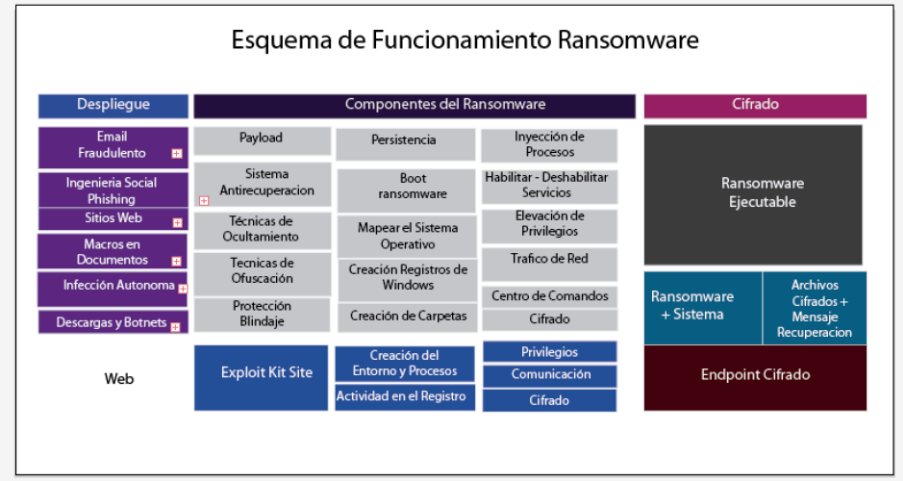
Por último, una vez generadas las llaves se requieren enviar hacia un Centros de Comandos y Control (C&C), que es el centro donde se opera y pueden enviarse las instrucciones u otros datos al *malware* que está en el equipo víctima. Aquí están algunas funcionalidades del servidor de C&C:

- El *malware* puede actualizarse con sus nuevas versiones desde el servidor de C&C.
- Algunos tipos de *ransomware* reciben la clave para cifrar desde el servidor de C&C.
- Usado en botnets para el control de equipos zombies.

Anteriormente, los servidores de C&C tenían IPs fijas y nombres de dominio que solían formar parte del *malware* . Lo que hizo más fácil para los proveedores de seguridad bloquear estas IPs y dominios usando firewall y detección de intrusos. Así, para evitar la detección, se empezaron a usar algoritmos para la generación de dominios, como DGA.

Teniendo en cuenta los conceptos y los comportamientos anteriores , en la Figura 2-2, se muestra el esquema de funcionamiento de un *ransomware*.

Figura 2-2.Esquema de Funcionamiento de un *ransomware*.



Fuente: Elaboración propia.

2.5 Formas de Distribución de un *Ransomware*

Es importante entender cómo se distribuye el *ransomware* para bloquear o prevenirlo. Si hay bloqueo desde la fuente que lo origina y se logra identificar el mecanismo de distribución del *ransomware* tendremos éxito en la detección y prevención contra el *ransomware.* El *ransomware* se distribuye de la misma manera que otros programas maliciosos. Estas técnicas de origen o distribución se denominan como vectores de ataque. Algunas formas de distribución son spam, phishing, sitios web infectados, macros en documentos o mediante el uso de *exploits* para sistemas operativos vulnerables. A continuación, se describen las formas más comunes de distribución de *ransomware*:

2.5.1 Distribución por correo Electrónico

Son generalmente usados porque contienen códigos maliciosos en los adjuntos o en los enlaces del correo electrónico que pueden conducir a un sitio hospedando un *exploit kit*. Algunas técnicas incluyen spam mediante imágenes con enlaces a sitios de *malware* (Salvi & Kerkar, 2017) .

2.5.2 Distribución por Ingeniería Social

Usando técnicas como el *phishing* y el *scaming* que permite engañar a usuarios, esto implica el envío de vínculos o archivos adjuntos maliciosos a la cuenta de correo de la víctima como por ejemplo enlaces de Dropbox o vínculos de redes sociales.

2.5.3 Distribución por Documentos con Macros

Los Ataques de Documentos con Macros son utilizados por los ciberdelincuentes mediante una macro que es un pequeño programa que puede ser incrustado en un documento de office o pdf (Trisha, 2015). La macro es un conjunto de comandos grabados que pueden ser ejecutados mediante un atajo de teclado o un clic. Para la creación de macros se utiliza Visual Basic que es un lenguaje de programación utilizado para crear o editar macros para Microsoft Word, Excel y Powerpoint. Las macros también se pueden utilizar para descargar *malware*. Para analizar documentos con macros maliciosas, OfficeMalScanner es la herramienta que permite realizar esta función.

2.5.4 Distribución a través de sitios web y/o Navegadores

Los ciberdelincuentes pueden realizar ataques web, son usados y aprovechados mediante una vulnerabilidad en los sitios web y en los navegadores. Una vulnerabilidad en una aplicación web, página web, base de datos o el servidor web pueden exponer el sitio web a ataques. Los atacantes pueden utilizar estas vulnerabilidades para comprometer el sitio web, estos pueden obtener las credenciales de los usuarios que han iniciado sesión en el sitio web. Además, un atacante puede incrustar código en las páginas web del sitio, el código incrustado tiene urls que pueden redirigir a la víctima a sitios maliciosos que pueden contener *ransomware* u otro *malware*, estas vulnerabilidades son del lado del servidor.

También hay vulnerabilidades del lado del cliente, mediante los navegadores web que tienen la capacidad de analizar el código en las páginas web alojadas en los sitios web y mostrarlo al usuario.. Una vulnerabilidad puede estar presente en el navegador o en su plugin lo que permitiría a un ciberdelincuente aprovechar la vulnerabilidad para él envió de código malicioso a la víctima.

2.5.5 Distribución por ataques automaticos

Estos permiten ejecutarse de manera autónoma buscando vulnerabilidades en un sistema operativo para ello utiliza un Exploit Kit, que es un programa que comprueba la versión del sistema operativo, las versiones de navegador, los plugins de navegador, las aplicaciones instaladas y las actualizaciones en el equipo, en consecuencia, sirve para detectar si hay una vulnerabilidad y ejecutar el exploit para luego explotar el equipo víctima. Una vez explotado el equipo, se comienza la infección mediante un downloader (descargador) que es un *malware* que puede ser configurado para descargar cualquier otro tipo de *malware*. Uno de los downloader más usados es Bede (Bridges, 2008) usado por algunos kits de exploit para la descarga de *ransomware*. Un ejemplo de ataque automático fue Crysis y Wannacry.

2.5.6 Distribución a través de Botnets

Las Botnets (Sanatinia & Noubir, 2015) también son usadas para el envió de *ransomware* mediante un centro de comandos y un downloader para descargar el *ransomware* y otros tipos de *malware*.

Esto genera una gran distribución de *ransomware*, una de las redes de bots más conocidas era Necrus que esparcía el *ransomware* Scrab y Locky.

Teniendo las formas de distribución , en la Figura 2-3, se muestra el diagrama de distribución o vectores de ataque de un *ransomware*.

Figura 2-2-3. Clasificación de vectores de ataque de un *ransomware*.

Fuente: Elaboración propia.

2.5.7 Ciclo de Vida de un Ataque de *Ransomware*

Algunos autores mencionan de varias formas el ciclo de vida, taxonomía y anatomía del *ransomware* y para efectos de este de trabajo se homologan todos los 3 conceptos a ciclo de vida de ataque de *ransomware*.

El ciclo de vida son las fases como el *ransomware* ataca y las acciones que realiza en cada una de ellas; se deben entender las estrategias, las tácticas y las técnicas que usan para poder generar una metodología de detección y prevención que cubra todo el ciclo de vida en sus diferentes fases.

Inicialmente, el ciberdelincuente necesita hacer un Despliegue del *ransomware,* para eso utiliza los vectores de distribución más apropiados para ejecutar el ataque, como la web, el correo, el uso de la red, una aplicación infectada o agregando en el *malware* la propiedad de ser una infección autónoma mediante una vulnerabilidad en el equipo. Esta fase también tiene un elemento importante llamado Entrega, que es el mecanismo de entrada del *ransomware*.

El *ransomware*, una vez descargado y ejecutado comienza la Instalación y Propagación; en esta fase el crypto *ransomware* inicia el payload para infectar el registro de Windows, instala las librerías necesarias y agrega las tareas automáticas para su funcionamiento.

En este punto, el *ransomware* utiliza los mecanismos de explotación e infección como los RAT y los Exploit Kit para infectar mediante una vulnerabilidad en el equipo, y poder llevar a cabo todo el proceso de comunicación con el centro de control y el cifrado de la información. Algunos de los Exploit Kit utilizados para generación de *ransomware* son Angler, usado para la creación de Locky; Neutrino, mediante el cual se generó Cerber y por último Rig (Vadim Kotov; Fabio Massacci, 2013).

Una vez instalado, el *ransomware* escanea y busca el contenido, en esta etapa el *ransomware* ya ha comenzado a buscar archivos y documentos tanto localmente como desde la red. Muchos ataques de *ransomware* priorizaron los recursos compartidos de red sobre los recursos locales (Aziz, 2016a).

Después de instalarse, escanear archivos y agregar las rutinas en los registros el *ransomware* requiere una comunicación hacia su Centro de Comando y Control para que puedan tomar control sobre el equipo**,** esta tarea se hace antes de comenzar o finalizar el ataque en el sistema. El *ransomware* se contacta a un servidor con el propósito de transferir las llaves de cifrado y garantizar si el ataque fue exitoso, con esto los ciberdelincuentes logran saber la cantidad de host y estaciones infectadas para lograr diferenciarlos mediante códigos únicos con el fin de realizar la extorsión más exitosa y poder diferenciar a quien enviarle las llaves de cifrado cuando la víctima realice el pago. Además, el cliente y el servidor de *ransomware* se autentican para validarse y dicho proceso es llamado "handshake". El servidor genera entonces dos claves criptográficas. Una clave

se guarda en el ordenador de la víctima y la segunda se guarda en el servidor del delincuente. (Zorabedian, 2015).

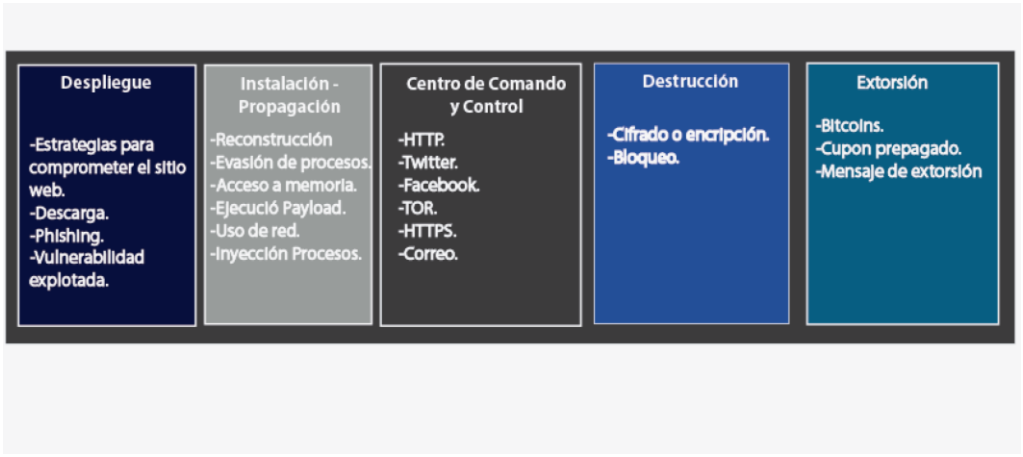
El proceso de Destrucción de un *ransomware* se realiza sobre el contenido de las carpetas, documentos y archivos, tanto en el host como a nivel de la red. Algunos *ransomware* tienen la propiedad de eliminar copias de seguridad como las Shadow Copies que almacenan copias de restauración del sistema, por lo que no deja al sistema posibilidad de restaurarlo a un estado anterior.

Por último, la Extorsión donde se genera una nota en la pantalla de la víctima con un límite de tiempo para pagar el rescate antes de que los ciberdelincuentes eliminen las llaves de descifrado, estos esperan hasta que se realice dicho pago.

Para comprender las fases de ciclo de vida de un ataque de *ransomware* se utilizó la metodología propuesta por (Aziz, 2016) y complementada con (Salvi & Kerkar, 2017) .

Teniendo en cuenta el ciclo de vida de un *ransomware* , en la Figura 2-3, se muestra el diagrama del funcionamiento de un ataque de un *ransomware*.

Figura 2-4. Ciclo de vida de ataque de un *ransomware*



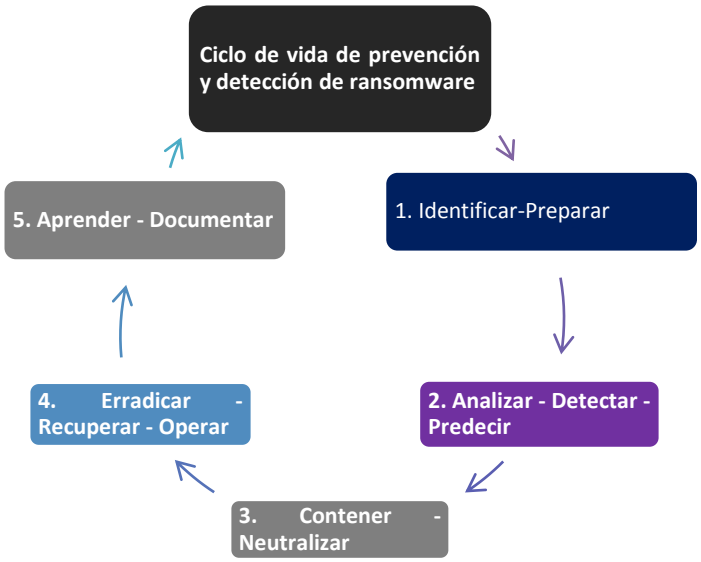
Fuente: Adaptada del sitio web Lynda.com.

2.6 Ciclo de Vida para la Detección y Prevención de *Ransomware*

Brewer (2016), plantea para el manejo de incidentes de *ransomware,* el marco de referencia de manejo de incidentes de la NIST 800-61, con el siguiente ciclo de vida: preparación, detección, contención, erradicación y recuperación, omitiendo la fase de análisis, la cual es la tarea más importante a la hora de entender el comportamiento de un *ransomware*. Para la elaboración del ciclo de vida para la detección y prevención de *ransomware,* se usó el ciclo de vida de respuesta de incidentes de la NIST 800-61, debido a que es el marco de referencia para prevenir, detectar, priorizar y gestionar incidentes.

Para efectos de este trabajo de grado, se realizaron algunos ajustes a la norma NIST 800-61, para poder aplicarlo sobre cualquier tipo de *malware* o un determinado *ransomware,* agregando la fase del componente de predicción al final de la fase de análisis y detección.

Figura 2-5. Ciclo de vida de prevención y detección de *ransomware*.



Fuente de elaboración propia.

Para iniciar se debe tener la identificación de los activos de información con el fin de tener una preparación de la arquitectura de seguridad mediante el uso de herramientas de software, infraestructura, hardware, equipos y la documentación que pueda permitir tener un panorama sobre el diseño de la plataforma de información o sistema en el cual operan las redes. Durante esta fase se crean los procesos y procedimientos que puedan servir para controlar las amenazas generando prevención sobre los activos de información.

Para la fase de preparación se debe contar con entrenamientos en el tema de análisis de *malware* ejecutando el procedimiento más adecuado en busca de una solución que mitigue el daño generado. Además toda actividad sospechosa debe quedar registrada en bitácora para posteriormente analizar y si es necesario determinar una investigación más exhaustiva en las comunicaciones de red por indicios que se hayan encontrado en otros sistemas como firewalls, proxys, entre otros.

Para los controles de prevención de *malware* es recomendable tener software antivirus para desplegarse en todo el sistema y organización. La protección contra *malware* debe implementarse a nivel de host, por ejemplo, servidores, estaciones de trabajo, sistemas operativos, a nivel del servidor de aplicaciones, servidor de correo electrónico, proxis y entre otros (Grance, Kent, & Kim, 2004).

Una vez se identificaron las amenazas y se categorizaron los tipos de incidentes se procedió a crear los mecanismos de detección que dieron la posibilidad de analizar las evidencias creadas por los registros en los sistemas de gestión de eventos, una vez se detectaron las amenazas se tomó acciones de control en el sistema debido que los diferentes tipos de incidentes merecen diferentes estrategias de respuesta. Para el manejo de incidentes una forma que se propone en este trabajo de grado es la predicción mediante el cual se puede evitar un incidente con antelación conociendo los patrones de comportamiento previos a un ataque como por ejemplo, si envían masivamente correos con *ransomware* mediante enlaces de Dropbox a un determinado grupo de personas dentro de una organización, se puede predecir que en algún momento el ciberdelincuente tiene la intención de acceder o cifrar la información. En este punto se deben concentrar los esfuerzos en detectar, analizar e identificar el *malware* y sus características, apoyándose en diferentes herramientas identificadas en la fase de preparación.

Para la determinación de las características del *ransomware* fue necesario realizar un estudio sobre las acciones que realiza una vez ejecutado el *malware* mediante una muestra en un entorno de virtualización o haciendo un análisis forense a un equipo infectado para encontrar las acciones o cambios realizados en el sistema de archivos, registros, procesos, y comunicaciones de red entre otros componentes del sistema. Además, se utilizarán sistemas automatizados tanto internos como en línea para complementar el análisis realizado y corroborar los datos obtenidos.

Una vez se identificó el tipo de ataque y el vector de ataque se deben generar las medidas de contención que puedan neutralizar las amenazas y mitigar el daño generado en la información. La contención es el paso más importante debido a que un incidente sobrecargue los recursos o aumente el daño. La mayoría de los incidentes requieren contención, por lo que es una consideración importante al principio del manejo de cada incidente. La contención proporciona tiempo para desarrollar una estrategia de remediación. Una parte esencial de la contención es la toma de decisiones (por ejemplo, apagar un sistema, desconectarlo de la red, desactivar ciertas funciones). Tales decisiones son mucho más fáciles de tomar si existen estrategias y procedimientos predeterminados para contener el incidente. Las organizaciones deben definir los riesgos aceptables al tratar con incidentes y desarrollar estrategias en consecuencia.

Las estrategias de contención varían según el tipo de incidente. Por ejemplo, la estrategia para contener una infección de *malware* transmitida por correo electrónico es muy diferente a la de un ataque DDoS basado en la red. Las organizaciones deben crear estrategias de contención separadas para cada tipo de incidente grave, con criterios claramente documentados para facilitar la toma de decisiones (Grance et al., 2004).

Una vez se contenga y neutralice el incidente se deben eliminar los potenciales archivos, registros o procesos que sean sospechosos de ser creados por el *malware*, garantizando que no se reactiven esperando una orden desde una central de comandos. Además, es importante documentar claramente cómo se han preservado todas las pruebas, incluyendo los sistemas comprometidos y las pruebas deben recopilarse de acuerdo con procedimientos, antes de eliminar los archivos o registros se deben dejar copias para análisis posteriores.

Después de eliminar la amenaza producida por un malware o un incidente de seguridad se procede con la recuperación permitiendo que se restauran los sistemas para que vuelvan a funcionar normalmente, y se deben buscar nuevas vulnerabilidades para evitar incidentes similares. La recuperación puede implicar acciones tales como restaurar sistemas a partir de copias de seguridad limpias, reconstruir sistemas desde cero, reemplazar archivos comprometidos por versiones limpias, instalar parches, cambiar contraseñas y reforzar la seguridad del perímetro de la red. Hay que tener en cuenta que una vez que un recurso es atacado exitosamente, es atacado de nuevo, u otros recursos dentro de la organización son atacados de manera similar.

Para finalizar en la respuesta a incidentes y en el análisis de *malware* también se debe aprender y mejorar. Debido a que los incidentes evolucionan reflejándose en nuevas amenazas, la tecnología mejora y también llegan nuevas amenazas por lo que se deben comprender estas amenazas.

Las lecciones aprendidas nos dan la experticia para tratar las nuevas formas de amenazas minimizando el impacto de cada una de ellas.

2.7 Métodos de Detección y Prevención para *Ransomware*

En la prevención y detección de *ransomware*, se busca desarrollar un esquema metodológico basado en la NIST 800-61 o NIST 800-83 sobre la gestión de incidentes de *malware*. Se comenzó con los métodos de protección existentes que sean capaces de contener y eliminar los patrones de comportamiento generado por las variables de *ransomware* seleccionadas. Una de las formas más efectivas para la detección y prevención es la ejecución de *ransomware* mediante entornos controlados y el uso de herramientas forenses para realizar un análisis estático o dinámico del comportamiento del *ransomware*, además de ingeniería inversa determinando las características más comunes y obtener resultados para tener una mayor comprensión de estos. A partir de ahí se construirá un modelo que relacione los métodos existentes para la detección y prevención de *ransomware* que agrupen las características más importantes para la contención y erradicación del *ransomware* sobre el ciclo de vida de detección y prevención de *ransomware (Mell, Kent, & Nusbaum, 2005).*

Debido a las estrategias y métodos que pueden ser definidas y puestos en práctica para protegerse de los *ransomware*, las estrategias que se presentan en las siguientes subsecciones sirven para la defensa y protección contra potenciales *ransomware*.

Con base en lo anterior, en este capítulo vamos a considerar los métodos de detección y prevención de *ransomware* más comunes tanto teóricos como prácticos para la formulación del esquema metodológico.

A continuación se explicarán los métodos usados para la elaboración del esquema metodológico.

2.7.1 Definición de Arquitectura de Seguridad

Una arquitectura de seguridad de la información es definida para las decisiones de negocio y mediante una política de seguridad de la información se pueden proteger contra los ataques de *malware* (Abhijit Mohanta, Mounir Hahad, Kumaraguru Velmurugan, 2018). En muchos casos, la arquitectura de seguridad se describe como una topología de red que refleje la tecnología de seguridad de la información.

El propósito de una arquitectura de seguridad de la información empresarial (EISA) es abordar un enfoque holístico de la seguridad de TI para garantizar que la seguridad de la información sea de manera coherente y con un nivel consistente de riesgo sobre los puntos críticos de la organización. Estará en su aplicabilidad en el negocio y su usabilidad por parte de los usuarios para protegerse contra sofisticado *malware*. Además, Tiene por objeto ayudar a tomar decisiones relacionadas con la identificación, adquisición, diseño, aplicación, implementación, despliegue y operación de elementos del entorno técnico de la organización.

2.7.2 Evaluación de Vulnerabilidades

La gestión de vulnerabilidades identifica los puntos débiles y las vulnerabilidades que tiene una organización en su infraestructura . Los productos de evaluación de vulnerabilidades exploran las condiciones vulnerables basándose en base de datos de vulnerabilidades conocidas, así como puertos abiertos, servicios y protocolos en ejecución, aplicaciones y sistema operativo. Esta información proporciona al área de seguridad los datos que necesitan para medir las posturas de

seguridad. Cuando el área de seguridad documenta la debilidad de la infraestructura de red y host, puede empezar a tomar decisiones sobre cómo eliminar la causa raíz de la mayoría de los ataques, reducir los vectores de ataque potenciales y limitar el impacto de un incidente de seguridad (Mark Nicolett & Amrit T. Williams, 2005).

2.7.3 Gestión de Parches de Seguridad

Actualizar el sistema operativo y las aplicaciones ayudan a proteger contra ataques de *malware* y *ransomware* debido a que en algunos casos se aprovechan de las vulnerabilidades en el sistema operativo como en el caso de Wannacry, las actualizaciones automáticas corrigen errores o fallas en el sistema y se debe tener cuidado con aplicaciones como Adobe Flash, Microsoft Silverlight y navegadores web (Christopher M Frenz & Christian Diaz, 2018).

2.7.4 Honeypot File

Una forma simple para la prevención para *ransomware* es mediante la creación de un honeypot de archivos la cual es propuesta por Chris Moore (Moore, 2016), el método consiste en la creación de carpetas falsas con el software tripwire o mediante Varonis DatAdvantage para que el *ransomware* interactúe con el software del sistema operativo y este determine actividad anormal; la desventaja de las carpetas señuelo es que no hay garantía de que el *ransomware* solo invada estas carpetas sino por el contrario cifre todo el sistema, y por lo tanto eludir la defensa.

El método propuesto del Honeypot File gira en torno a la recopilación de información sobre un ataque y su uso para la defensa, el estudio demostró que el honeypot file puede identificar al usuario que infectó el sistema junto con el volumen de los archivos que son modificados, además de informar las acciones mediante alertas de correo electrónico o un sistema de gestión de eventos (Moore, 2016).

Otro método propuesto mediante honeypot es (Cabaj & Mazurczyk, 2016), dicho método consiste en la creación de un entorno llamado Maltester; el concepto de funcionamiento es mediante máquinas virtuales y añadiendo un software de gestión de eventos, responsable de controlar el proceso de análisis. Cuando un usuario añade una nueva muestra al sistema (llamada interfaz remota), la máquina atacada o de destino crea una instantánea a partir del estado actual. Esta

máquina (que actualmente ejecuta Windows XP) tiene un software que interactúa con el host de gestión; así, la muestra de *malware* se transfiere automáticamente al sistema destino y es ejecutado. En este instante, se inicia el análisis dinámico sobre la máquina atacada para determinar el tipo de infección y el tipo *ransomware*. Este método es funcional solo para entornos virtualizados y no representa una solución para la detección y prevención de *ransomware*, pero contribuye para el esquema metodológico propuesto en este trabajo de investigación.

2.7.5 Indicadores de compromiso (IoC)

Un IoC son piezas de datos forenses que permiten en una red o en un sistema operativo indicar una intrusión informática. Los IOCs típicos son firmas de virus y direcciones IP, hash MD5 de archivos de *malware* o urls o nombres de dominio de servidores de control y comando de botnet. Después de que los IOCs hayan sido identificados en un proceso de respuesta a incidentes e informática forense, pueden ser usados para la detección temprana de futuros intentos de ataque usando sistemas de detección de intrusos y software antivirus (Nate Lord, 2017).

2.7.6 Análisis de Comportamiento de Usuario (UBA)

Una UBA permite identificar actividades sospechosas basándose en la forma en que el usuario interactúa con los sistemas y el entorno. Su estilo de escritura, los comandos más utilizados, la rutina matutina, las aplicaciones que utilizan, la velocidad a la que trabajan y muchas otras facetas pueden ser analizadas para identificar el comportamiento normal del usuario frente al comportamiento anormal del usuario. Muchas organizaciones víctimas de *ransomware* como Cryptolocker, CTB Locker y otros utilizan el análisis de comportamiento para que mitigar estos ataques así como otros como son el abuso de información privilegiada y credenciales comprometidas (Jeromie Jackson, 2016).

(Anton Chuvakin, 2017) proponen un método de análisis de la conducta de usuarios usado especialmente en el ámbito de las amenazas internas. Un UBA es una forma única de detectar, alertar y posiblemente bloquear un ataque de *ransomware*. En lugar de buscar conductas específicas, la UBA se basa en determinar qué es lo que está fuera normal en un sistema o una red Por ejemplo, es inusual que una aplicación acceda a todos los archivos de un sistema de manera

rápida, pero eso no significa que este comportamiento no ocurra. Podría ser un programa de respaldo accediendo a esos archivos.

2.7.7 Coincidencia de Patrones

Un método para la prevención de *ransomware* es la creación de firmas mediante la coincidencia de patrones, estos incluyen patrones de cadenas en texto y cadenas binarias complejas. Para escribir firmas para la detección de cadenas binarias es necesario entender los conceptos de ingeniería inversa, depuración y tener una buena comprensión del lenguaje ensamblador.
Una firma de *malware* puede estar compuesta de hashes, cadenas legibles y cadenas binarias. Para poder usar la coincidencia de patrones usaremos la herramienta YARA utilizada para identificar *malware* en este caso *ransomware* (Duc, 2016).

2.7.8 Firmas mediante HASH

Un hash es un número único que se utiliza para identificar un dato. Los algoritmos de reducción como MD5, SHA1 y SHA2 son usados para identificar el hash de los datos. Los algoritmos hash pueden comprobar que un archivo no se haya modificado por lo tanto es único. Si se calcula un hash del archivo completo (todos los datos dentro del archivo), no se parecerán a otro archivo (Preneel, 2010).

2.7.9 Aprendizaje de Maquina

El aprendizaje de maquina o automático es la optimización de un modelo de predicción para describir mejor el mapeo a partir de los datos de entrada a su etiqueta de destino asignada, con el objetivo final de poder predecir la etiqueta de destino de datos desconocidos. La optimización del modelo se denomina alternativamente model fitting o model training (Nieuwenhuizen, 2017)..

Para el contexto de la detección de *ransomware*, el aprendizaje automático es la formación o creación de un algoritmo de decisión para reconocer ciertos rasgos de comportamiento (los datos de entrada) de los procesos en ejecución que discrimina entre el *ransomware* y las aplicaciones benignas no maliciosas (la etiqueta de destino).

El entrenamiento del algoritmo de decisión requiere tres componentes: el conjunto de datos de entrenamiento que contiene ejemplos de *ransomware* conocidos (es decir, etiquetados) y aplicaciones benignas conocidas, la captura del comportamiento de manera cuantificable, y el esquema de clasificación que define la formación y el aprendizaje del algoritmo de predicción (Nath & Mehtre, 2014) .

Un método sobre aprendizaje de máquina es EldeRan, propuesto por(Sgandurra, Muñoz-González, Mohsen, & Lupu, 2016), es mediante un enfoque de aprendizaje automático para clasificar el *ransomware* basado en sus primeras acciones. Los *ransomware* tienen características dinámicas únicas y, para detener su propagación, es crucial detectar nuevas variantes durante su primera aparición. Para ello, EldeRan selecciona en primer lugar, las características relevantes que caracterizan el *ransomware* y, a continuación, clasifica cada una de las nuevas instalaciones en un PC de usuario a través de un algoritmo de aprendizaje de máquina, para llevar a cabo la detección, sin confiar en las clasificaciones mediante las técnicas heurísticas o basadas en firmas. El objetivo de EldeRan es entender si se puede identificar el *ransomware*, con un alto grado de precisión, utilizando un número limitado de rasgos característicos y antes de infectar a las víctimas. Además, EldeRan proporciona una forma automática de crear firmas para nuevas variantes de familias de *ransomware*. EldeRan complementa bien los mecanismos basados en firmas de AV, ya que se puede utilizar para identificar casos en los que las AV pueden haber pasado por alto familias de *ransomware* nuevas o desconocidas.

2.7.10 Antivirus

Es un software que tiene como fin detectar o eliminar un *malware*. El proceso de un antivirus se inicia cuando un fichero llega a un sistema, incluso antes de que el fichero sea escrito en el disco, el antivirus lo detecta y comienza el análisis en tiempo real. Después de eso, el motor llama la función de escaneo de archivos para escanear el archivo contra las firmas que tiene. Si la firma coincide, el archivo se elimina o se pone en cuarentena. El antivirus puede escanear todo el sistema de archivos y procesar virtualmente memoria. Si identifica *malware*, toma medidas contra él. Esto se llama limpieza. El software antivirus puede tener los siguientes motores de escaneo: Analizador de archivos, Escáner de memoria, Desempaquetador, Detector de rootkits, Limpieza del motor

Los motores pueden recuperar datos a través de algoritmos que pueden identificar el patrón en los datos. Los algoritmos comunes aplicados a estos datos son los siguientes:

- Algoritmos Hashing: SHA1, SHA2, MD5
- Algoritmos de coincidencia de patrones

Las firmas escritas en los antivirus para *malware* se pasan a través de estos algoritmos. La firma puede ser única para un archivo o puede detectar varios archivos. Una única firma que puede identificar varios archivos con contenidos diferentes esto se llama firma genérica. A veces el antivirus tiene firmas que dicen que un archivo es sospechoso, pero no puede confirmar la maldad, estas firmas se denominan heurística.

2.7.11 Antivirus Escáner de Archivos

Un escáner de archivos es una de las características más importantes de un motor antivirus, tiene la capacidad de identificar varios formatos de archivo y analizarlos para recuperar más datos, como el tipo de archivo y tipo de extensión; en palabras sencillas, podemos decir que un escáner de archivos puede realizar análisis estático en un archivo (Sgandurra, Muñoz-González, Mohsen, & Lupu, 2016).

Todos los archivos, incluidos los ejecutables, tienen propiedades estáticas. Las propiedades estáticas de un archivo ejecutable son aquellas que se pueden visualizar sin ejecutar el fichero. El ejecutable de Windows PE tiene las propiedades estáticas que se pueden ver usando muchas herramientas. CFF Explorer es una herramienta que puede ayudar a explorar las propiedades estáticas.

2.7.12 Anti-Ransomware Tools

Es una herramienta que se diferencia de un antivirus porque es más ligera para analizar y bloquear el *ransomware* y software malicioso de cifrado de forma inmediata. El software *anti-ransomware* más conocido es de Kaspersky *Anti-Ransomware* y el *Anti-Ransomware* Tool de Bitdefender (P Tailor & Patel, 2017).

Los *anti-ransomware* generan una defensa adicional contra el *ransomware*, están diseñados para ejecutarse en segundo plano y bloquear los intentos de cifrado de datos del *ransomware*. También supervisan el registro de Windows en busca de cadenas de texto que se sabe que están asociadas con *ransomware*. El problema con este enfoque es que necesita instalar software cliente en cada dispositivo de red (Darragh Delaney, 2016).

2.7.13 Anti-bootkit

Una medida de protección contra ataques bootkit en el proceso de arranque es la utilización del sistema basado en arranque seguro con interfaz de firmware extensible unificada (UEFI), este funciona distinto al proceso de BIOS convencional porque añade la protección contra la infección del bootkit a partir de Windows 8, también incorpora mejoras respecto a la limitación de número y tamaño de particiones, así como otras medidas de seguridad. Para prevenir la modificación del sector de arranque podemos utilizar la herramienta MBRFilter (Christopher M Frenz & Christian Diaz, 2018).

2.7.14 Prevención de Ejecución de Datos (DEP)

Para protegerse contra el desbordamiento de la pila o stack, usado por algunas variantes de *ransomware*, Windows ha creado un concepto llamado Data Execution Prevention (DEP), esto evita la ejecución de estas áreas (Liu & Kuhn, 2010).

2.7.15 Segmentación de Red

Una forma de aislar los ataques de *ransomware* es mediante la segmentación de la red con la creación de VLAN y Listas de Control de Acceso (ACL). Este método sirve para controlar el tráfico que hay entre las diferentes redes de la organización; si bien la segmentación no previene ataques de *ransomware,* sí ayuda a minimizar que la infección no se propague por toda la red, sino en un determinado segmento específico de red (Xiyang & Chuanqing, 2009).

2.7.16 Entornos de Virtualización

Una forma de controlar los ataques *ransomware*, es mediante la ejecución de instantáneas sobre las máquinas virtuales que se encuentren en entornos virtualizados, esto con el fin de lograr que, si un *ransomware* cifra la información, se pueda volver al estado actúa u original de la máquina virtual, la falencia de este control es que, si un *ransomware* daña el archivo de las instantáneas, no habría forma de volver al estado original de la máquina virtual (Kao, Chi, & Lee, 2014).

2.7.17 Cortafuegos

Los cortafuegos son componentes de red que monitorean y controlan tanto el tráfico entrante como el saliente. Actúan como una barrera entre la red interna -que protegen- y la externa, llamada red no confiable. Los cortafuegos pueden variar desde el filtrado de paquetes, que funciona en la capa 3 hasta el filtrado de aplicaciones capa 7; las capas 3 y 7 se refieren a las capas de un OSI modelo de redes (Trabelsi & Molvizadah, 2016).

También existen cortafuegos que pueden ser implementados en un equipo de escritorio o host; estos se denominan *firewall* basado en host, estos también tienen la posibilidad de añadir reglas para bloquear accesos hacia direcciones ip de la red TOR mediante blacklist o listas negras.

2.7.18 Sistemas de Detección de Intrusos

Un sistema de detección de intrusos (IDS) y sistemas de prevención de intrusos (IPS), son componentes de red que monitorean el tráfico de red en busca de actividades maliciosas, o bien sólo tiene que detectar y alertar al usuario del tráfico malicioso en caso de IDS, o bloquearlo de forma proactiva en el caso de IPS. Al igual que los cortafuegos, los IDS e IPS también están basados en filtrado de paquetes básicos, para prevenir anomalías en el protocolo de capa 3 y capa 4 y analizar las cargas útiles del paquete.

Al igual que los cortafuegos también tienen motores que identifican protocolos de la capa 7 y supervisar las aplicaciones que se ejecutan en la parte superior de los protocolos de la capa 7, caso Facebook o Instagram (Vasanthi & Chandrasekar, 2011).

Los IDS/IPS son basados en firmas, aunque los nuevos módulos y componentes tienen análisis de grandes datos impulsado por anomalías y aprendizaje de automático para detectar actividades de *malware*

2.7.19 Sandbox

Los sandbox son utilizados para capturar el comportamiento de un archivo y, a continuación, asociarlo con *malware* o eliminarlo. Un sandbox es un sistema automatizado de análisis de *malware*, Cuckoo es uno de los proyectos de sandbox de código abierto (Segu-Info, 2016).

Un sandbox básico puede producir los siguientes datos para una muestra proporcionada:

- Motor de análisis estático
- Motor de análisis de comportamiento

Un motor estático, como su nombre indica, puede mostrar las propiedades estáticas del archivo, en el caso de propiedades de Windows PE. Un motor de comportamiento puede mostrar datos que se relacionan con el comportamiento. El comportamiento puede incluir registros de API, archivos de conexión de red y cambios en el registro.

2.7.20 Gestión de Eventos de Información de Seguridad - SIEM

Proporcionan un análisis en tiempo real de las alertas de seguridad generadas por el hardware y software de red. Las soluciones SIEM pueden venir como software, appliance, o administración de servicios, y también son utilizados para validar datos de seguridad y generar reportes para fines de complimiento. Además, están dedicadas al escaneo activo y pasivo de sus redes para detectar eventos y actividades sospechosas, y predecir los ataques antes de que tengan lugar. Evitando ataques informáticos y mejorando la seguridad (Mark Nicolett & Amrit T. Williams, 2005).

2.7.21 Filtrado de contenido de correos o Antispam

Mediante el filtrado de correos contenido se puede generar una protección basada en la reputación de correo electrónico o bloquear archivos ejecutables con extensión .exe, .zip o url sospechosas. Los mensajes potencialmente dañinos pueden bloquearse antes de que el destinatario tenga la

oportunidad de abrirlos. Algunas soluciones antispam son Symantec Email Gateway Security o Sophos Email Exchange (Symantec, 2018).

2.7.22 Filtrar extensiones .JS peligrosas

Se pueden filtrar los enlaces con extensión .js que pueden ser peligrosas debido que contienen (ejecutables y scripts) y se pueden bloquear a través de listas negras o mediante la comprobación de reputación del sitio web mediante la herramienta https://transparencyreport.google.com/safe-browsing de google (Segu-Info, 2016).

2.7.23 Cambiar Sistemas Operativos y protocolos obsoletos

Usar sistemas operativos y protocolos obsoletos permite que un atacante pueda vulnerar el sistema de manera más rápida infectando a toda una organización ,un ejemplo fue el *ransomware* Wannacry que infectó mediante el protocolo smbv1 y smbv2 a máquinas con sistema operativo Windows XP y 7. Para prevenir el ataque de un *ransomware* es necesario realizar un cambio de las versiones del sistema operativo y actualizar a protocolos de seguridad más robustos (Christopher M Frenz & Christian Diaz, 2018).

2.7.24 Políticas de restricción de software (SRP)

Las políticas de restricción de software (SRP) permiten o prohíben el lanzamiento de archivos ejecutables mediante una política de grupo local o de dominio. Los métodos de protección contra virus o *ransomware* que utilizan SRP sugieren prohibir la ejecución de archivos desde directorios específicos del entorno de usuario, a los que suelen llegar los archivos o archivos de *malware*. En la mayoría de los casos, los archivos que contienen un virus se obtienen de Internet o del correo electrónico y se guardan en el directorio (%APPDATA%) y en el perfil de usuario y las carpetas (%Temp%) Las copias temporales de archivos comprimidos descomprimidos también se almacenan en este directorio (Tim Buntrock, 2016).

2.7.25 Windows File Services Resource Manager (FSRM) - Bloquear las extensiones de archivos creadas por ransomware

El servicio de Windows File Services Resource Manager (FSRM) permite bloquear automáticamente las actividades de *ransomware*. Los archivos cifrados de *ransomware* tienen extensiones tales como *.a19 *.a5zfn *.aaa *.abc *.adk *.adr pero mediante FSRM se puedan bloquear este tipo de extensiones para que no modifiquen y tengan acceso al sistema (Spiceworks, 2016).

2.7.26 Mostrar las extensiones de archivos ocultos

Los *ransomware* ocultan sus nombres de archivo para engañar a las víctimas para que los ejecuten,si el nombre de archivo del *ransomware* es realmente factura.pdf.exe, la víctima sólo puede verlo como factura.pdf, debido a que Windows no muestra el archivo ejecutable oculto que se presenta en formato pdf de forma predeterminada (Christopher M Frenz & Christian Diaz, 2018).

2.7.27 Aumento de los archivos renombrados

Cambiar los nombres de los archivos no es una acción común cuando se trata de actividad en archivos compartidos en red. En el transcurso de un día normal, se puede terminar con sólo conjunto de archivos de renombrados, incluso si tiene cientos de usuarios en su red. Solo cuando el *ransomware* ataca, resultará un aumento masivo de los archivos renombrados a medida que sus datos se cifran (Darragh Delaney, 2016).

2.7.28 Lista blancas o negras de todas las aplicaciones

Se pueden crear listas blancas sólo con las aplicaciones que sean de confianza y ejecutadas en la organización, y una lista negra con las aplicaciones que no se requieran ejecutar en estaciones de trabajo, esto con el fin de minimizar el uso de las aplicaciones potencialmente dañinas.

2.7.29 Bloquear la reproducción automática de medios extraíbles

Es recomendable desactivar la ejecución del software desde un medio extraíble para evitar que procesos dañinos no se inicien automáticamente desde medios externos, como memorias USB u otras unidades de disco (Tripwire, 2016).

2.7.30 Bloquear ventanas emergentes o popups

Instalar un complemento de navegador puede bloquear las ventanas emergentes, ya que también pueden ser un punto de entrada para ataques de *ransomware* (Tripwire, 2016). Algunos bloqueadores de popups o ventanas emergentes son Adblock y Adblock Plus y pueden ser instalados mediante GPO en entornos Windows (Christopher M Frenz & Christian Diaz, 2018).

2.7.31 Deshabilitar la ejecución de archivos temporales de instalación

Por lo general el *malware* y el ransomware cuando inicia la fase de ejecución accede a modificar archivos y carpetas tales como %AppData, %Program Data% y %Temp% , un método de prevención es bloquear los permisos de escritura para que estos no puedan sobrescribir los procesos o archivos de ejecución del sistema (Christopher M Frenz & Christian Diaz, 2018).

2.7.32 Configuración de Escritorio Remoto Seguro

Mediante el protocolo de Escritorio Remoto (RDP) un atacante puede vulnerar el sistema mediante ataques de contraseñas o diccionario debido al uso de contraseñas débiles , una vez logra identificar la contraseña el atacante puede cifrar los datos o archivo con *ransomware.* Un método para la prevención de ataques de RDP es usar cifrado SSL o TLS en la conexión mediante doble autenticación mediante NAC (Christopher M Frenz & Christian Diaz, 2018).

2.7.33 Deshabilitar Macros

Las macros en los documentos de Office son uno de los vectores de ataque más utilizado por los ciberdelincuentes, se sugiere bloquear las macros en los documentos de Office para que no se ejecuten en archivos desde Internet (Krunal & Viral, 2017).

Para analizar documentos con macros es recomendable usar la herramienta **OfficeMalScanner** que permite escanear documentos de Office en busca de macros ocultas antes de abrirlos y poder determinar si el documento contiene actividad maliciosa. Las macros sólo pueden ocultarse y ejecutarse automáticamente dentro del formato de documento de tipo Office 2007 con las

extensiones DOC, XLS, PPT. Para los nuevos formatos de archivo DOCX, XLSX y PPTX no pueden ejecutarse de manera automática y haciendo que las macros sean incompatibles (Trisha, 2015).

2.7.34 Deshabilitar servicio de Volume Shadow Copy

El servicio de vssadmin.exe es una funcionalidad en Windows para administrar el Servicio de instantáneas de volumen, la herramienta se puede usar para restaurar versiones de Windows anteriores en caso de daños o *malware*. Sin embargo, el *ransomware* puede cifrar archivos de las instantáneas de volumen, vssadmin.exe se convierte en un problema en lugar de un servicio funcional. Si está deshabilitado el servicio de vssadmin en un equipo en el momento de un ataque por medio de *ransomware* no podrá usarlo para borrar las instantáneas de volumen. Esto significa que puede usar vssadmin.exe para restaurar los archivos cifrados posteriormente (Tripwire, 2016).

2.7.35 Recuperación de archivos

Para la recuperación de los archivos cifrados depende de la variante de *ransomware* y de la versión, a medida que ha avanzado las variantes de este tipo de *malware* se han ido perfeccionando, los desarrolladores de *malware* han corregido los errores y han agregado nuevas funcionalidades como algoritmos de cifrado más avanzados y mejoras en la forma de comunicación ver Anexo 3-1. Para la recuperación de *ransomware* algunos fabricantes de antivirus han generado herramientas que permiten la recuperación tales como Rakhni Decryptor, Rannoh Decryptor, Bitdefender Anti *Ransomware* Remove Tools y Trend Micro Anti *Ransomware* Tools (Christopher M Frenz & Christian Diaz, 2018) ver Anexo 3-2. Herramientas de Recuperación *Ransomware*.

2.7.36 Cifrado de Archivos

Mediante el cifrado de archivos mediante la herramienta Bitlocker para los sistemas operativos Windows que permiten proteger el arranque y los archivos de sistema. Así que puede ser una prevención contra el *ransomware*, debido a que encripta el sistema, documentos y fotos para cuando el *ransomware* se ejecute le solicitara autenticación y permisos de escritura y ejecución.

2.7.37 Copias de Seguridad en Nube o Disco Externo

Una forma de prevención son las copias de seguridad para proteger los datos, no sólo contra *ransomware*, sino también contra los daños físicos que pueden producirse en los dispositivos de almacenamiento y en los equipos de una organización. Es importante para la prevención del *ransomware*, pero además contra cualquier tipo de amenaza, como por ejemplo daños en el disco duro o físico. Se puede hacer una copia de seguridad mediante el almacenamiento en nube como Amazon Drive, Google Drive o OneDrive para mejorar la seguridad en las copias de seguridad o mediante discos externos.

2.7.38 Prevención de Pérdida de Datos (DLP)

Es una herramienta de software que puede detectar el robo de datos. Un DLP trabaja monitoreando el uso, movimiento y almacenamiento de datos confidenciales. Las soluciones de DLP se utilizan para lo siguiente: Prevención de fugas de datos y Prevención de pérdida de datos.

La fuga de datos y la pérdida de datos son términos estrechamente relacionados. Cuando no podemos recuperar los datos, los llamamos pérdida de datos. Cuando los datos confidenciales llegan a personas no autorizadas, se llama filtración de datos. Cuando los datos están encriptados por el *ransomware*, en algunas ocasiones es irrecuperable. Esto puede denominarse perdida de datos. El robo de datos puede considerarse como fuga de datos.

Los datos en uso pueden filtrarse en los puntos finales mediante USB, correos electrónicos, FTP, etc. Los datos en movimiento pueden ser un filtró a través de la red o en los correos electrónicos de los empleados y el tráfico de la red generado por el *malware*. Los DLP se basan en el control de acceso, es decir, quién debe acceder a qué datos y los permisos que tiene el usuario.

2.7.39 Copias del Archivo Shadow Copy

Shadow Copy es una herramienta de Windows que permite realizar copias de seguridad de forma manual o automática mediante instantáneas de archivos o volúmenes, incluso cuando están en uso. Si realizamos una copia del archivo mediante *shadow copy* y lo exportamos a una copia externa podemos regenerar el sistema a partir de la copia. Por lo general la gran mayoría de los *ransomware*

eliminan el archivo del shadow copy para la restauración del sistema debido a que se encuentra almacenado en el sistema operativo, pero al estar en un disco externo se tiene la posibilidad de recuperar el sistema mediante la copia de *shadow copy*.

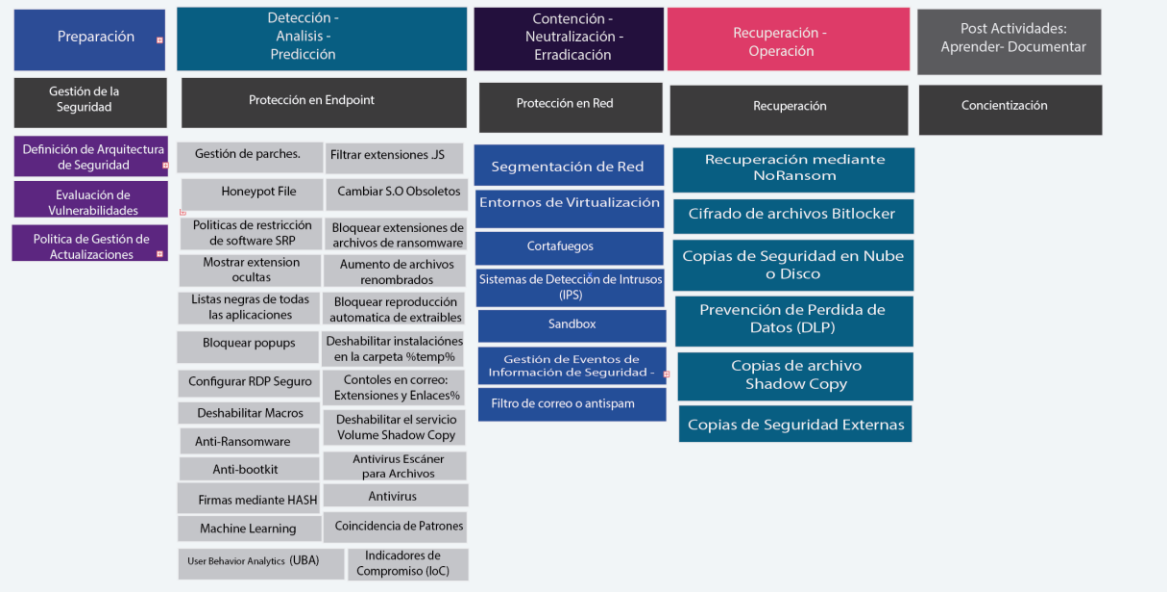
2.7.40 Concientización

Los usuarios finales son una de las principales amenazas de una organización (Rebecca Wilson, 2017) debido a los comportamientos inadecuados y la falta de información sobre las amenazas existentes .Una forma de lograr prevenir y detectar ataques de *ransomware* es mediante la concientización y entrenamiento de los usuarios finales sobre los riesgos a los cuales están expuestos dentro una organización, por lo cual se deben generar simulaciones en ingeniería social para entrenar a los usuarios finales sobre los riesgos a que están expuestos en la organización y así puedan notificar al área de seguridad de las actividades sospechosas que se les generen (Christopher M Frenz & Christian Diaz, 2018).

2.8 Esquema Metodológico de Detección y Prevención de *Ransomware* Propuesto

Basados en los métodos anteriores, y para efectos para el presente trabajo de grado se propone el siguiente esquema metodológico para la detección y prevención de *ransomware*.

Figura 2-6. Esquema metodológico para la detección y prevención de *ransomware* en una estación de trabajo.



Fuente de elaboración propia.

2.9 Esquema de medición CVSS

El Sistema Común de Puntuación de Vulnerabilidades (CVSS) es un marco abierto para comunicar las características y la gravedad de las vulnerabilidades del software. El CVSS consta de tres grupos métricos: Base, Temporal y Ambiental. El grupo Base representa las cualidades intrínsecas de una vulnerabilidad que son constantes en el tiempo y a través de los entornos de usuario, el grupo Temporal refleja las características de una vulnerabilidad que cambia con el tiempo, y el grupo Medioambiental representa las características de una vulnerabilidad que son únicas del entorno de un usuario. Las métricas Base producen una puntuación que va de 0 a 10, la cual puede ser modificada mediante la puntuación de las métricas Temporal y Ambiental. Una puntuación CVSS también se representa como una cadena vectorial, una representación textual comprimida de los valores utilizados para derivar la puntuación (first.org, 2019)

En consecuencia, la CVSS ha entregado una tabla de valoración acorde a los niveles de criticidad (en su versión 3), que se ilustra en la siguiente figura:

Figura 2-7. Escala para la calificación de severidad.

Rating	CVSS Score
None	0.0
Low	0.1 - 3.9
Medium	4.0 - 6.9
High	7.0 - 8.9
Critical	9.0 - 10.0

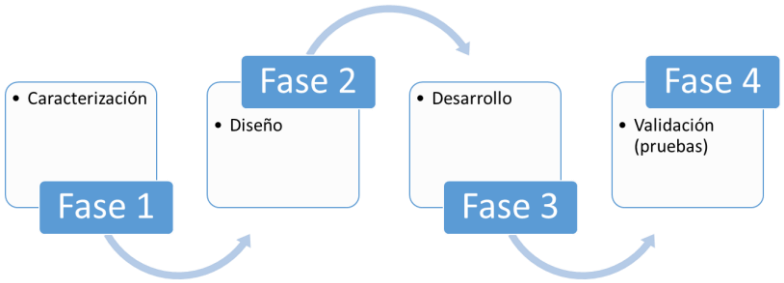
Fuente: CVSS v3, en línea en: https://www.first.org/cvss/specification-document#1-2-Scoring

3. METODOLOGÍA

El éxito en la defensa contra estos ataques de *malware* depende principalmente de la comprensión del entorno, además de saber cuáles son los activos críticos que podrían vulnerar los ciberdelincuentes mediante cualquier tipo de técnica, y más aún si logran ejecutar un *ransomware;* esto con respecto de cómo construir un esquema metodológico que permita generar seguridad de la información, partiendo de la comprensión del *ransomware* y la forma como evoluciona, para saber el daño que puede causar y así aumentar la preparación y generar controles más efectivos para combatir dichas amenazas.

Para el desarrollo del proyecto, se abordaron diferentes fases a saber:

Figura 3-1: Fases para el desarrollo del proyecto.



Fuente de elaboración propia

A continuación, se describe cada una de las fases y cómo se abordó:

3.1 Fase 1: Caracterización

En esta fase de caracterización, se tomaron 24 familias de Ransonware (anexo 2-2), estas familias se consideran de mayor impacto en los sistemas informáticos, dichas familias se seleccionaron de múltiples fuentes de acuerdo a la evolución del malware (como ya se indicó en el marco teórico), esto es, aquellos malware que a través del tiempo han tenido una evolución más relevante y acotado a los sistemas Windows.

Para las familias seleccionadas, se valoraron 23 comportamiento que pueden ser identificados a la hora de infectarse, éstos comportamientos fueron obtenidos de diferentes fuentes y que puedan dar cuenta de la existencia de un malware en el sistema. Estos comportamientos son:

1. Modificación y cifrado de archivos
2. Realiza actividades en el registro de windows
3. Crea directorios para su funcionamiento
4. Notificación de ransomware
5. Actividad en Redes
6. Eliminación de Shadow Copies
7. Autoeliminación después de ejecución
8. Geolocalización IP
9. Búsqueda de antivirus y firewall
10. Modificaciones en MBR
11. Deshabilita iniciar en modo seguro
12. Terminación de procesos
13. Bloquea el escritorio
14. Reemplaza el tapiz de escritorio
15. Deshabilita la restauración del sistema
16. Deshabilita las actualizaciones automaticas
17. Deshabilita el servicio de Windows Security Center
18. Terminación de Administrador de tareas
19. Deshabilita la restauración de Windows
20. Deshabilita el Windows Error Reporting
21. Deshabilita el Antivirus y el Firewall
22. Apaga el equipo una vez infectado
23. Busca todas las unidades de disco y de red

La estrategia de caracterización, consiste es identificar acorde al malware, cual es el comportamiento o los comportamientos que se identifican según su operación o forma de actuar en el sistema. Cada uno de estos se evalúo y se marcó dependiendo del comportamiento identificado, luego, se hizo una sumatoria de comportamientos que poseen.

Con el resultado final (sumatoria), se obtuvo el porcentaje posible de afectación, en consideración que el mayor puntaje es de 23 (es el caso en que un malware tenga todos los comportamientos), éste será el 100%. En consideración con la homologación de la tabla de medición de la CVSSv3 (descrita en el marco teórico), se realizó un extrapolación del porcentaje obtenido de las tablas (Anexo 2.2) y la CVSS, seleccionado aquellos considerados por encima del 39%, esto es, lo que son considerados medio, alto y crítico. Se ha seleccionado la escala CVSS dado su nivel de madurez, su fortaleza en la medición y la asociación directa con la afectación en las plataformas de manera técnica.

En la siguiente tabla (3-1) se muestra la extrapolación de los valores con los cuales se obtuvo los códigos maliciosos elegibles:

Tabla 3-1. Extrapolación de valores para medir los malware a elegir: CVSS vs. Medición propia.

Medición CVSS			**Medición Propia**	
puntuación	%	Nivel	% características	Cantidad de características seleccionadas
0	0%	Nula	0%	0
0,1 - 0,39	10% - 39%	Baja	10% - 39%	3-8
4,0 - 6,9	40% - 69%	Media	40% - 69%	9-15
7,0 - 8,9	70% - 89%	Alta	70% - 89%	16-20
9,0 - 10,0	90% - 100%	crítica	90% - 100%	21 - **23**

Fuente: Autores a partir de las escala de CVSS

3.2 Fase 2: Diseño

Para el diseño del esquema metodológico y en consideración de la selección en la fase 1, se reunieron todos los diferentes métodos de detección y prevención para *ransomware,* mediante un análisis cuantitativo y cualitativo de la información recolectada en las diferentes bases de datos académicas y repositorios de conocimiento. Este esquema metodológico se caracterizó principalmente por la implementación de los controles y métodos más pertinentes para la

prevención y detección de *ransomware* en una estación de trabajo, dicha información está basada en los referentes teóricos de autores y revistas que nos permiten la formulación de un esquema metodológico; esto quiere decir que se hizo una valoración por medio de una revisión documental los métodos existentes para la prevención y detección de *ransomware* y a partir de ahí, nos permitió desarrollar un esquema con los diferentes métodos y apoyado en los elementos de las normas NIST 800-61 y 800-83, para el manejo de incidentes relacionados con *malware.*

Una vez se definieron los métodos, fueron valorados en el Anexo 5-1 Validación del Esquema Metodológico con Métodos de Detección y Prevención de *Ransomware* para así crear el esquema metodológico ver Figura 2-6.

Los métodos de detección (D) y prevención (P) fueron valorados para cada uno de los ransomware hallados en la fase 1, dichos métodos son: Controles generales, host, Controles de defensa perimetral y en red, Políticas o Controles, Recuperación y Concientización.

Cada uno de los métodos de valoró en características individuales así:

Controles generales

Definición de una arquitectura de seguridad

Evaluación de la vulnerabilidad

Host

Gestión de parches

Honeypot File

Indicadores de compromiso (IOC)

User Behaviour Analytics

Coincidencia de patrones

Firmas mediante HASH

Machine Learning

Antivirus

File Scanner Antivirus Engine

Anti-Ransomware

Anti-bootkit

Data Execution Prevention (DEP)

Controles de defensa perimetral y en red

Segmentación de Red

Entornos de virtualización

Cortafuegos

IPS

Sandbox

SIEM

Filtrado de contenido de correos o Antispam

Políticas o Controles

Filtrar extensiones .js

Cambiar Sistemas Operativos y protocolos obsoletos

Políticas de restricción de software (SRP)

Bloquear las extensiones archivos de ransomware

Mostrar las extensiones de archivos ocultos

Aumento de los archivos renombrados	
Lista negra de todas las aplicaciones	**Recuperación**
Bloquear autoplay o la reproducción automática	Recuperación de archivos mediante NoRansom
Bloquear Popups	Cifrado de archivos Bitlocker
Deshabilitar ejecución de archivos temp instalación	Copias en Nube o en disco
	Prevención de Pérdida de Datos (DLP)
Configuracion de Escritorio Remoto Seguro	Copias del archivo Shadow
Documentos: Deshabilitar Macros - OfficeMalScanner	**Concientización**
	Concientización
Deshabilitar servicio de Volume Shadow Copy	

Con el resultado obtenido, se hace el desarrollo de la herramienta para la validación de los controles (fase 3).

3.3 Fase 3: Desarrollo

Para su desarrollo, se caracterizaron por medio de revisión documental, un conjunto de herramientas para la prevención y detección de *ransomware,* y a partir de ahí se seleccionó una herramienta que permitió realizar un análisis sobre las distintas variantes de *ransomware,* identificando y analizando si las variantes son efectivamente un tipo de *ransomware*. Una vez se identificaron las herramientas, se procedió a la elección de una de ellas mediante la valoración del Anexo 4-1 Herramientas de Análisis de Malware, donde se determinó usar la herramienta de código abierto MALICE que permite adaptar y utilizar el lenguaje de programación Python haciendo uso de scripts personalizados, y el *plugin* YARA que permitió identificar los patrones de comportamiento en archivos ejecutables tipo *ransomware*. Las variantes fueron ejecutadas en un ambiente controlado mediante una herramienta de virtualización y la ejecución de comandos de consola de Windows, donde se ejecutaban los scripts de MALICE identificando si una variante de *ransomware* efectivamente era *ransomware.* El resultado de la ejecución de la consola de comando exportaba un archivo en formato .txt que contenía la información con el análisis y el resultado. Una vez teniendo los resultados en el archivo txt se procedió a realizar el desarrollo que interpretó los resultados de una manera más comprensible y clara; la herramienta desarrollada utilizó PHP y MySQL y permitía importar los resultados contenidos en el Anexo 4-6, donde se logró diseñar un panel de visualización para mostrar los resultados y lograr una mejor lectura y comprensión,

además sirvieron como material para identificar, detectar y prevenir un *ransomware*. También se diseñó una vista de resultados con gráficas para valorar y cuantificar la cantidad de *ransomware* analizados en una estación de trabajo.

3.4 Fase 4: Validación

Para la validación, se utilizó la estadística descriptiva con el fin de ~~para~~ determinar la sumatoria de métodos de detección y prevención utilizando las siguientes fórmulas:

La fórmula para la sumatoria de los Métodos de Prevención es la siguiente:

VTMP = $\sum MP$

Donde

VTP = Valoración Total Métodos de Prevención

$\sum MP$ = Sumatoria de Métodos de Prevención

La fórmula para la sumatoria de los Métodos de Detección es la siguiente:

VTD = $\sum MD$ / Total de variantes

Donde

VTD = Valoración Total Métodos de Detección

$\sum MD$ = Sumatoria de Métodos de Detección

Para el cálculo de los porcentajes de detección y prevención se debe tener en cuenta la sumatoria de métodos dividido por el total de métodos que en total fueron 40. Para ello se utilizaron las siguientes fórmulas:

Para el Porcentaje de Prevención:

P_{TMD} = $\sum MP$ / Total de Métodos de Prevención x 100

Donde

P_{TMP} = Porcentaje Total Métodos de Prevención

$\sum MD$ = Sumatoria de Métodos de Prevención

Para el Porcentaje de los Métodos de Detección:

P_{TMD} = $\sum MD$ / Total de Métodos de Detección x 100

Donde

P_{TMD} = Porcentaje Total Métodos de Detección

$\sum MD$ = Sumatoria de Métodos de Detección

4. RESULTADOS

4.1 Caracterización de Ransomware

En consideración con la metodología ya estipulada, y con el fin de construir un desarrollo para la detección y prevención para *malware* tipo *ransomware* se realizó el proceso de ingeniería inversa a un conjunto de muestras de código fuente de diferentes *ransomware* procedentes de los repositorios de GitHub, Gitlab y Bitbucket. Dicho proceso de ingeniería inversa permitirá entender el uso, la estructura y el funcionamiento de algunas variantes de este *malware*.

Para cada familia se obtuvo la información respectiva de las diferentes características en su comportamiento (ver anexo 2-1), luego de analizar dicho comportamiento, se realizó la calificación.

El análisis que se realiza a continuación se centra en los criterios asociados a cada una de los vectores del *ransomware* y las características tales como método de entrega, tipo de extensión, formato de cifrado, eliminación de copias de restauración, comunicación con centro de comandos, servicio de descifrado, pago y público objetivo (Salvi, Salvi, & Kerkar, 2017). Dichas fases serán usadas para entender la anatomía o ciclo de vida del *ransomware* para consolidar y construir un esquema metodológico de detección y prevención de *ransomware*.

Para desarrollar la caracterización se tomaron 24 muestras de *ransomware* que son:

1. AIDS
2. Archievus
3. Reveton
4. Cryptolocker
5. Locker
6. CryptorBit
7. Synolocker
8. CryptoBlocker
9. CTB-Locker
10. Onion
11. CoinVault
12. CryptoWall
13. OphionLocker
14. TeslaCrypt
15. VaultCrypt
16. CryptoWall 2.0
17. CryptoWall 3.0
18. CryptoWall 4.0
19. Locky
20. Cerber
21. Crysis
22. Dharma
23. Wannacry
24. Petya

Una de las tareas principales fue realizar un estudio cronológico donde se determina el inicio o punto de partida para llegar a la actualidad. En este primer paso se notó que todas las familias analizadas exhiben comportamientos similares o iguales, evolucionando tanto en el método de cifrado como los vectores de infección, los cuales serán detallados en el Anexo 2-1. Familias de *Ransomware*.docx.

A partir de las 24 muestras seleccionadas se categorizaron los criterios más importantes para realizar el análisis, los resultados obtenidos demuestran que el *malware* tipo *ransomware* genera capacidad al evolucionar, una de las características más importantes que tienen los *ransomware* es que partir de la creación de nuevas tecnologías mutan hacia ella, es decir si aparece una nueva forma de tecnología como el Cloud este lograra operar en ella o se aprovechara de dicha tecnología buscando mimetizarse para atacar, se debe comprender que al iniciar también nuevas tecnologías como por ejemplo IoT se deben también generar controles y alarmas que permitan detectar nuevas formas de *ransomware* que puedan poner en riesgo cualquier tipo de infraestructuras.

Al ser un sistema operativo tan complejo, en especial los Windows y con la variedad de procesos que tiene, se deben comprender que elementos pueden ser susceptibles a fallos o vulnerabilidades que puedan poner en riesgo al sistema operativo, como por ejemplo si tomamos las muestras y analizamos los síntomas de los *ransomware* tales como el Cryptolocker usado mediante correo electrónico, CTB-Locker agregando comunicación con un centro de comandos para evitar la captura de las llaves, Wannacry se aprovecha de la vulnerabilidad MS17-010 en los Windows, Crysis aprovecha vulnerabilidades en RDP, Petya modifica el MBR y la tabla de particiones, los vectores de despliegue y ejecución son distintos por la forma de operación en el sistema operativo, se logra identificar que la mayoría acceden al registro de Windows para realizar modificaciones al sistema.

En algunos casos hay similitudes en cuanto a funcionamiento y operación, excepto cuando se trata de modificaciones de versiones futuras, como por ejemplo el CryptoWall que tiene 4 versiones cada una de ellas con una modificación mejor a la anterior.

Además de comprender los vectores de entrada, que permiten el despliegue de un *ransomware* por parte de un atacante, y la forma cómo lograría llegar a una estación de trabajo para su posterior

ejecución e interacción por parte de un usuario, si bien ya hay mecanismos que permiten la detección y prevención del *ransomware* como los antivirus, firewall o ids, no hay una metodología integrada que cubra todas las fases desde el despliegue, ejecución, centro de comandos, destrucción y estrategias para la no extorsión. Los términos nombrados anteriormente los llamaremos el ciclo de vida del *ransomware*.

En términos de la evolución del *ransomware,* tenemos que entender la dirección en la cual van a migrar las nuevas variantes del *ransomware,* y bajo esta perspectiva analizar qué tipo de servicios y tecnologías pueden ser blanco de ataques, una vez se identifiquen cuáles son esos tipos de servicios y tecnologías se deben buscar los mecanismos, estrategias o metodologías que permitan integrarse y alinearse a la solución propuesta en esta tesis.

Por último, una de las formas de utilización del *ransomware* puede ser la usada por los hackers una vez explotado un sistema, este lograría cifrar la información de forma manual o automática y no dejar evidencias o registros haciendo más difícil el análisis forense, además de la pérdida de información.

Como resultado final se seleccionaron 12 muestras:

Id	Familia
9	CTB-Locker
12	CryptoWall
14	TeslaCrypt
15	VaultCrypt
16	CryptoWall 2.0
17	CryptoWall 3.0
18	CryptoWall 4.0
20	Cerber
21	Crysis
22	Dharma
23	Wannacry
24	Petya

Ahora bien, de estas 12 muestras, en el análisis realizado, se encontró que los malware *CryptoWall, CryptoWall 2.0, CryptoWall 3.0 y CryptoWall 4.0* poseen comportamiento similar, por lo cual, solo se analizó ***CryptoWall.***

4.2 Diseño de Herramienta de Prevención y Detección de Ransomware en Estación de Trabajo

Se definieron los requisitos del diseño de la arquitectura basados en las herramientas MALICE y YARA. Para realizar el diseño se requieren los indicadores de compromiso (IoC) de cada uno de estas variantes y donde se utilizó la herramienta Any.Run para extraer dichos indicadores ver Anexo 4-2. Indicadores de Compromiso *Ransomware*. La arquitectura para el diseño está basada en un equipo de escritorio, tal y como se observa en la Figura 4-1 mediante un entorno de virtualización mediante VMWARE y el conjunto de herramientas seleccionadas de detección y prevención. Cada *ransomware* se evaluó de manera independiente con el fin de conocer con mejor detalle los comportamientos que impactan al equipo y para una mejor interpretación de los resultados, esto debido a que ejecutar varias muestras de *ransomware* en el equipo alterarían el sistema de manera impredecible y se haría más improbable tener resultados más acertados sobre la validación y valoración de la herramienta. La herramienta puede interactuar con el *ransomware* ejecutando un análisis de varias muestras al mismo tiempo mediante MALICE Y DOCKER con cada firma de antivirus y para este trabajo de grado tomaremos cada variante con el fin de tener mejores conclusiones y resultados.

En la siguiente figura 4-1, se puede ilustrar los resultados de la valoración de los diferentes controles con respecto a los Malware ya seleccionados:

Figura 4-1. Resultados del esquema metodológico de detección y prevención de ransomware.

Fuente de elaboracion propia

Para el total de metodos de prevención y detección se tomaron en cuenta todos los comportamientos de las 23 variantes de ransomware seleccionadas con el fin de crear un esquema metodologico más completo y no sobre las 8 variantes seleccionadas.

Para el total de métodos de prevención , que son los metodos que permiten avisar o aletar sobre el impacto negativo que causa un ransomware, se encontró que los 40 métodos de prevención frente a cada una de las variantes de ransomware tuvieron un porcentaje de prevencion es del 100%, es decir todos los metodos propuestos sirven para prevenir, esto debido a que dichos métodos de prevencion, puede ser cualquier control que se tome de manera anticipada para evitar que suceda la ejecución de un ransomware y representa el grueso de la muestra.

Para el total de metodos de detección que son los que realmente hacen el control de contener y proteger de manera efectiva se encontro que de los 40 metodos para deteccion, solo 12 métodos que pueden detectar un ransomware en una estación de trabajo, esto debido a que, los métodos no tienen la capacidad de bloquear o impedir la ejecución de un ransomware y representa la tercera parte de la muestra con el 30%.

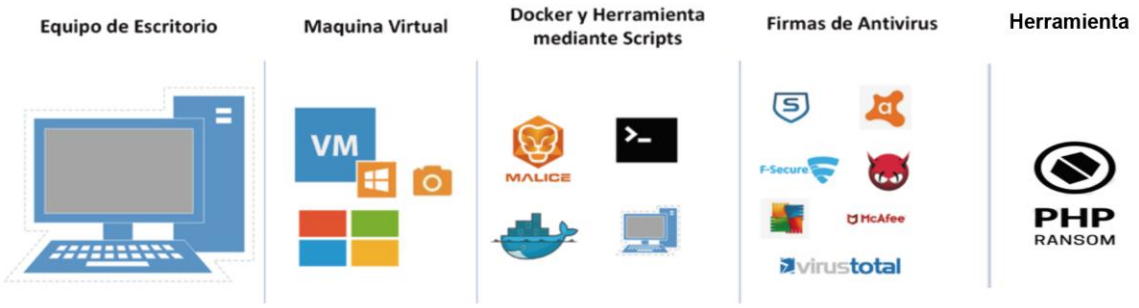
En consecuencia, se evaluaron en total 40 métodos diferenciando los metodos de prevencion y de detección, los cuales indican que 40 metodos sirven para realizar prevención (el 100% de los controles) y solo 12 sirven para realizar detección (el 30% de los controles evaluados).

Con los datos anteriores, se hizo el siguiente diseño de la arquitectura (figura 4-1) la cual comprende los siguientes elementos: Implementación del desarrollo para detección y prevención de ransomware con las subfases que aparecen a continuación.

Selección de herramienta
Configuración del equipo para el desarrollo
Instalación y configuración de Malice y Yara
Estructura del Código
Simulación de la herramienta
Pruebas de validación
Análisis de resultados
Desarrollo de herramienta de visualización de resultados

Fuente: Elaboración propia.

Figura 4-2.Diseño de la arquitectura del entorno de herramienta.



Fuente: Elaboración propia.

4.3 Implementación del Desarrollo para Detección y Prevención de Ransomware

Los ransomware que no se utilizaron fueron debido a que las muestras no se encontraron en los repositorios de analisis de malware y estos fueron los siguientes : Vaultcrypt, Darmha, Cryptowall 2,3,4 debido a que en el momento que se empezo a realizar este trabajo e investigacion eran muy recientes y no habia mucha informacion sobre en los repositorios como github o giblab. Estas muestras se cambiaron por Locky y Petya las cuales si se encontraron los repositorios.

4.3.1 Selección de Herramienta para la Detección y Prevención de *Ransomware* en una Estación de Trabajo

Como anteriormente se mencionó, se deben conocer las herramientas de prevención y detección de *ransomware*, para la cual se tiene que realizar el filtro con las correspondientes herramientas que permitan protegernos de un *ransomware* en una estación de trabajo. La herramienta desarrollada está adaptada y tomada de repositorios de códigos abierto para luego ser probada en un ambiente controlado, donde se realizarán pruebas de validación con cada una de las muestras seleccionadas y validando la efectividad de la herramienta contra el *ransomware*, además se generaron registros y anexos donde se muestre las acciones y el comportamiento que realiza el *ransomware* con cada una de las variables que fueron seleccionadas para así aplicar medidas o acciones en la herramienta desarrollada. Se tienen en cuenta los riesgos asociados en el entorno controlado con el fin de no causar alteraciones en otros sistemas.

Se inició con las definiciones de las herramientas que se consideran relevantes para la prevención y detección de *ransomware* teniendo en cuenta los atributos de funcionamiento, uso y tipo para saber cuál es la más apropiada para este trabajo de grado. Durante la evaluación de las herramientas se seleccionará una o varias como aparecen en el Anexo 4-1 Herramientas de Análisis de *Malware*, que sirven como guía para elegir la más pertinente y apropiada para el desarrollo de la detección y prevención de *ransomware* en una estación de trabajo mediante la categorización, clasificación y valoración de estas herramientas. Se determinó la implementación del Marco de Análisis de *Malware* Masivamente Escalable llamado MALICE debido a la valoración y puntuación obtenida en el Anexo 4-1. Herramientas de Análisis de *Malware*.

La herramienta seleccionada permite analizar *malware* mediante las diferentes marcas reconocidas de firmas de antivirus y se considera ser una versión gratuita de código abierto de Virus Total para que pueda ser usada por cualquier persona y a cualquier escala. MALICE también permite la creación scripts mediante YARA y el uso del lenguaje de programación Python debido a que es multiplataforma y es un lenguaje interpretado, esto significa que no se compila a diferencia de otros lenguajes como Java y C++, además es usado en Big Data, Inteligencia Artificial, Aprendizaje de Máquina y en seguridad informática para la creación de *exploit* y herramientas de *pentesting* y hacking ético (Manuel Zaforas, 2017). Con esto se logra hacer una detección y una prevención sobre los *ransomware* que fueron seleccionados en el capítulo 2.

4.3.2 Desarrollo de herramienta para la detección y prevención de *ransomware* en una estación de trabajo

Este capítulo tiene como objeto mostrar el desarrollo de la herramienta para la detección y prevención de *ransomware* en una estación de trabajo*,* se determinó usar la herramienta de código abierto MALICE (Marco de Análisis de *Malware* Masivamente Escalable) debido que permite adaptar y utilizar el lenguaje de programación Python y mediante el uso del *plugin* YARA que serán modificados para generar una herramienta de detección y prevención de *ransomware*. La herramienta analiza 8 de las variantes de *ransomware* que fueron seleccionadas en el capítulo 2, estas variantes serán ejecutadas en un ambiente controlado para determinar el comportamiento y establecer si mediante el desarrollo de la herramienta se permite prevenir y detectar el *ransomware* en la estación de trabajo*.* Los resultados obtenidos sirven como material para valorar, cuantificar, identificar, detectar y prevenir un *ransomware* en una estación de trabajo, logrando el cumplimiento el objetivo principal de este trabajo*.*

4.3.3 Implementación y Desarrollo de Herramienta de Prevención y Detección de *Ransomware* en Estación de Trabajo

Para el desarrollo de la herramienta tendremos un equipo con Sistema Operativo Windows 10 y la utilización de DOCKER, MALICE y YARA. Para iniciar la implementación y desarrollo tendremos 4 fases descritas donde se iniciarán a continuación:

A. Configuración del Equipo para el Desarrollo del Ambiente Controlado

A continuación se listan las características en hardware del equipo y el software utilizado para el desarrollo de la herramienta para la detección y prevención de *ransomware* para este trabajo de grado.

Tabla 4-1. Descripción de equipo para el ambiente controlado.

Descripción	Característica	Tipo de Herramienta
Procesador	Core I3 de 4ª Generación.	Hardware
Memoria	6 Gigas	Hardware
Tamaño del disco duro	500 Gigas	Hardware
Tarjeta de red	1 Giga	Hardware
Word	Procesador de texto	Software de edición
Excel	Hoja de cálculo	Software de edición
Sublime Text	Editor de texto	Software para edición
Python	Lenguaje de programación	Software de programación
MALICE	Analizador de malware	Software Múltiples Antivirus
YARA	Coincidencia de patrones	Software plugin de Python
Regshot	Revisión de registros	Software para monitoreo registros

Capture Bat	Revisión de registros	Software para monitoreo de registros
Process Explorer	Revisión de procesos	Software para Monitor de procesos
Any.run	Analizador de malware	Sandbox para IoC online
Wireshark	Capturar tráfico de red.	Software Analizador de red
IDA	Desensamblador	Analizar ejecutables.
Docker	Contenedor	Virtualización de aplicaciones

B. Instalación y Configuración de MALICE y YARA

MALICE es una herramienta de código abierto multiplataforma que permite analizar *malware* con múltiples antivirus y herramientas además de poderse instalar en S.O Windows, Linux y Mac OS, para su instalación en Windows descargamos la versión 0.3.24 de Github https://github.com/MALICEio/MALICE/releases ver Anexo 4-3. Instalación MALICE.

Una vez se finalice la instalación de MALICE iniciaremos la instalación Python 2.7 de Windows y se descarga de https://www.python.org/downloads/release/python-2715/ .

Para la instalación de YARA en Python usamos la consola de python y ejecutamos el comando i**mport YARA** ver Anexo 4-4. Instalación Python y YARA.

C. Estructura y Funcionamiento del Código

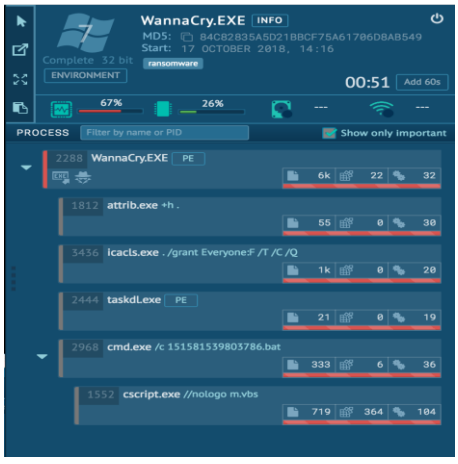
Es importante considerar que para el funcionamiento del código fue necesario realizar el análisis Estático y dinámico del código fuente de los r*ansomware*, en la Figura 4.2,donde se tomó la variante de *ransomware* Wannacry y se analizó mediante la herramienta IDA donde nos muestra la estructura de código y los procesos que puede ejecutar en el sistema, la estructura de código, se puede visualizar en lenguaje ensamblador ver Figura 4-3.

Figura 4-3.Análisis Estático de *ransomware* Wannacry mediante IDA

Fuente: Elaboración propia.

Para los indicadores de compromiso utilizamos la herramienta Any.run para analizar los diferentes comportamientos y patrones que genera el *ransomware* en un equipos con S.O Windows. En la Figura 4-3 se muestra la coincidencia del comando *cacls . /grant Everyone:F /T /C /Q* que permite mostrar o modificar las listas de control de acceso de archivos permitiendo alterar las carpetas y archivos. Con la información analizada de cada una de las variables y los comportamientos y donde genero un anexo de indicadores de compromiso para las variantes de ransomware seleccionadas Ver Indicadores de Compromiso Ransomware se iniciara la construcción de las reglas para la detección y prevención de *ransomware.*

Figura 4-4. Indicador de compromiso Wannacry.



Fuente: Elaboración propia.

Para la creación y desarrollo de las reglas de YARA, tomamos los Indicadores de Compromiso (IoC) suministradas en el Anexo 4-2. Indicadores de Compromiso *Ransomware* de las variables de *ransomware* seleccionadas en el capítulo 1. Este análisis permite crear las reglas que se observar en la Figura 4-4, para la detección y prevención de Wannacry. La regla incluye los comandos que ejecuta el *ransomware* cuando inicia el proceso de ejecución en una máquina.

Figura 4-5.Creación de regla en YARA para la detección.

```
rule WannaCry_Ransomware {
   meta:
      description = "Detects WannaCry Ransomware"
      author = "Florian Roth (with the help of binar.ly)"
      reference = "https://goo.gl/HG2j5T"
      date = "2017-05-12"
      hash1 = "ed01ebfbc9eb5bbea545af4d01bf5f1071661840480439c6e5babe8e080e41aa"
   strings:
      $x1 = "icacls . /grant Everyone:F /T /C /Q" fullword ascii
      $x2 = "taskdl.exe" fullword ascii
      $x3 = "tasksche.exe" fullword ascii
      $x4 = "Global\\MsWinZonesCacheCounterMutexA" fullword ascii
      $x5 = "WNcry@2ol7" fullword ascii
      $x6 = "www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com" ascii
      $x7 = "mssecsvc.exe" fullword ascii
      $x8 = "C:\\%s\\qeriuwjhrf" fullword ascii
      $x9 = "icacls . /grant Everyone:F /T /C /Q" fullword ascii
```

Fuente de elaboración propia.

D. Simulación de Herramienta en Estación de Trabajo

Una vez creadas las reglas para prevenir y detectar cada variante de *ransomware* y se procede a ejecutar el comando MALICE para el escaneo de las 8 muestras seleccionadas de *ransomware* y realizar las validaciones correspondientes de la efectividad del desarrollo del agente de prevención y detección.

Como se observa en la Figura 4-5, se inicia el proceso de ejecución de MALICE utilizando la herramienta DOCKER para la detección y prevención de *ransomware*, para la simulación se tomó cada una de las variantes y con cada una de las firmas de antivirus que permitan generar los resultados para su posterior validación. Para la simulación utilizamos los scripts de ejecución que se encuentra en la carpeta 4.6 Anexos Script de Ejecución.

Figura 4-6. Simulación de para detección y prevención de *ransomware*.

Fuente: Elaboración propia.

E. Pruebas de Validación

Para la validación se tomaron los resultados de la simulación de cada una de las variantes de *ransomware*, donde por cada firma de antivirus mediante Malice y cada variable de *ransomware* se marca con una X en caso de que la firma detecte la variante de *ransomware* analizada, ver Anexo 4-7 Resultados y donde se encuentran definidos con el nombre de cada variante ver Figura 4-7.

Figura 4-7.Resultados de la simulación de *ransomware* mediante MALICE.

Nombre	Fecha de modificación	Tamaño	Clase
CerberResult_Analisis.txt	24 oct 2018 11:50	46 KB	Texto
CryptolockerResult_Analisis.txt	26 oct 2018 8:30	16 KB	Texto
CryptowallResult_Analisis.txt	ayer 10:04	33 KB	Texto
CTB-LockerResult_Analisis.txt	24 oct 2018 9:33	25 KB	Texto
Locky-Result_Analisis.txt	ayer 9:44	16 KB	Texto
PetyaResult_Analisis.txt	ayer 10:40	45 KB	Texto
TeslaCryptoWallResult_Analisis.txt	ayer 9:55	19 KB	Texto
WannacryResult_Analisis.txt	ayer 10:34	16 KB	Texto

Para la validación, es necesario la construcción de los indicadores generados en los resultados que permitan analizar cada una las variantes teniendo en cuenta la efectividad que tiene cada firma de antivirus mediante la herramienta desarrollada contra cada una de las variantes de *ransomware.*

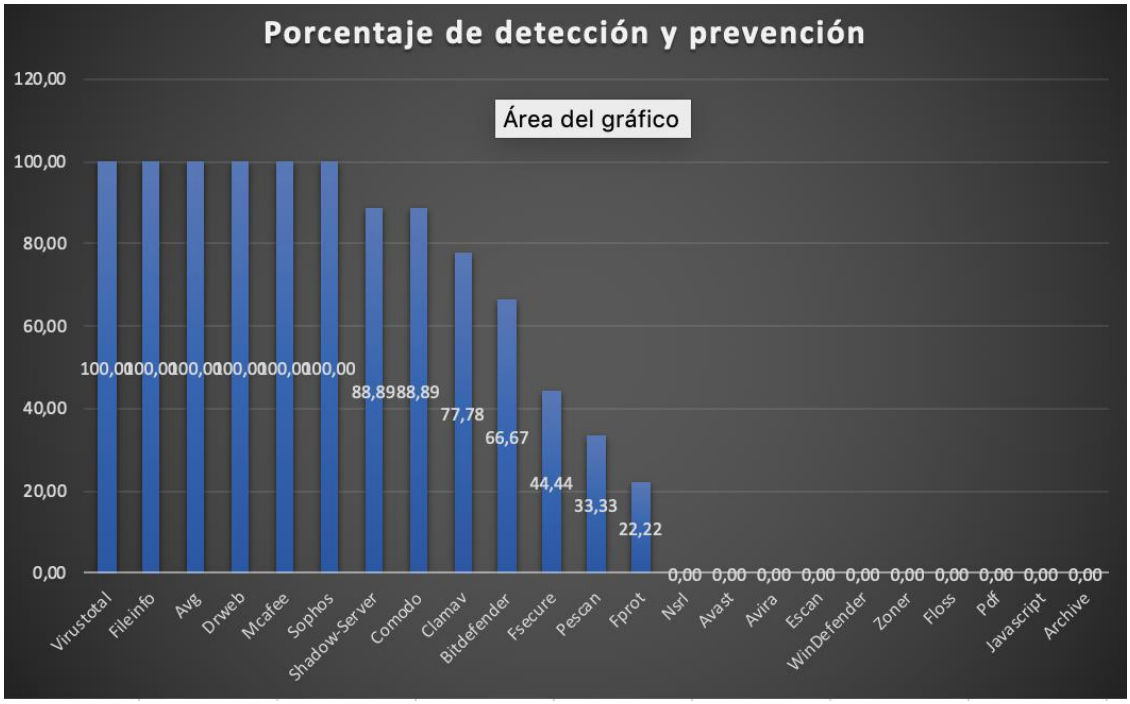
Tabla 4-2. Valoración y Evaluación de *Ransomware* con MALICE y YARA.

MALICE	CTB-Locker	CryptoLocker	Locky	TeslaCrypt	CryptoWall	Cerber	Wannacry	Petya
Nsrl	-	-	-	-	-	-	-	-
Virustotal	X	X	X	X	X	X	X	X
Shadow-Server		X	X	X	X	X	X	X
Fileinfo	X	X	X	X	X	X	X	X
Avast	-	-	-	-	-	-	-	-
Avg	X	X	X	X	X	X	X	X
Avira	-	-	-	-	-	-	-	-
Bitdefender	X	X	-	-	-	X	X	X
Clamav	X	-	X	-	X	X	X	X
Comodo	X	X	X	X	X	X	X	X
Drweb	X	X	X	X	X	X	X	X
Escan	-	-	-	-	-	-	-	-
Fprot	-	-	X	X	-	-	-	-
Fsecure	X	X	-	-	-	X	-	-
Mcafee	X	X	X	X	X	X	X	X
Sophos	X	X	X	X	X	X	X	X
WinDefender	-	-	-	-	-	-	-	-
Zoner		-	-	-	-	-	-	-
Pescan	X	-	-	X	-	-	-	-
Floss	-	-	-	-	-	-	-	-
Pdf	-	-	-	-	-	-	-	-
Javascript	-	-	-	-	-	-	-	-
Archive	-	-	-	-	-	-	-	-

F. Análisis de los Resultados Obtenidos

Para analizar los resultados obtenidos están basados en la valoración de la tabla 4-2 y se muestran de una manera más detallada en la Figura 4-7.

Figura 4-8. Porcentaje de detección y prevención de *ransomware*.



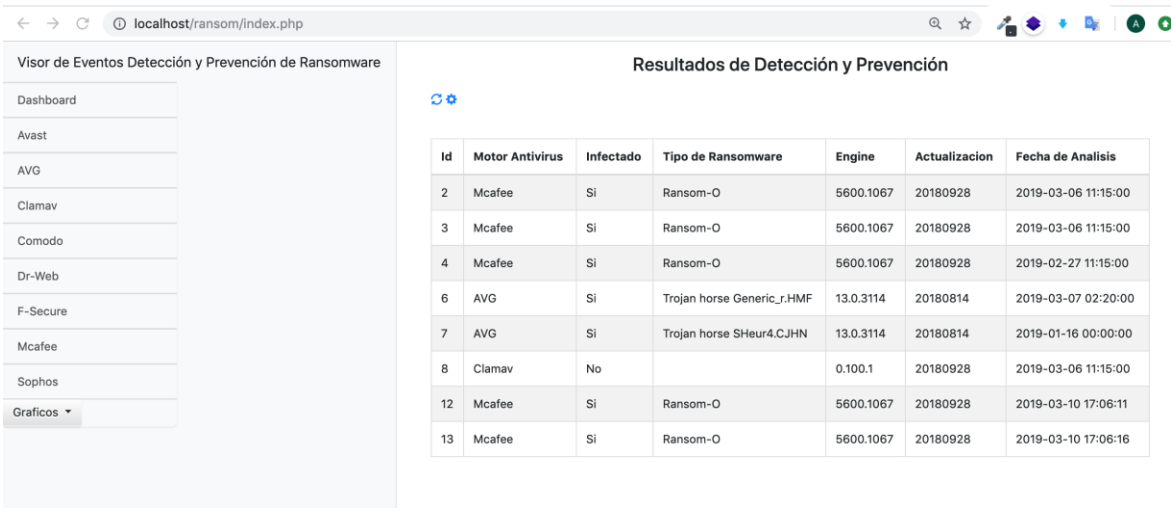
En este caso, mediante el uso de la herramienta MALICE se detectó que los sistemas de antivirus enlazados mediante firmas y scripts como Virustotal, AVG, Mcfee,DrWeb y Sophos tienen el 100% de detección y prevención Ver Anexo 4-5. Evaluación de Herramienta *Ransomware*, además de identificar que no solo suficiente con el uso de MALICE y YARA por lo que es necesario utilizar otras herramientas de software permiten complementar el desarrollo de la herramienta de prevención y detección de *ransomware* debido que el 43% de las herramientas totales detectan y puede prevenir de forma correcta, por lo que se requieren otro tipo de reglas que permitan tener un efectividad más precisa de los comportamientos del *ransomware* ver tabla 4-2.

G. Desarrollo de Herramienta para Visualizar los Resultados de Detección y Prevención de *Ransomware*

Una vez se tienen los archivos de los resultados ver sección 4.3.5, se proceden a ingresar los resultados mediante el desarrollo de una herramienta llamada PHP Ransom diseña y construida en lenguaje php y con un motor de base de datos MySQL, ver Figura 4-9, esto con el fin de comprender y visualizar los resultados de una manera más sencilla. La herramienta dispone de varias opciones

las cuales son: La opción dashboard es donde se visualizan los resultados después de realizar la carga de los archivos exportados con Malice, y para cada firma de antivirus se dispone de un conjunto de opciones con cada uno de los motores antivirus, por último está la opción Gráficos.

Figura 4-9. Herramienta de Detección y Prevención de *Ransomware*.

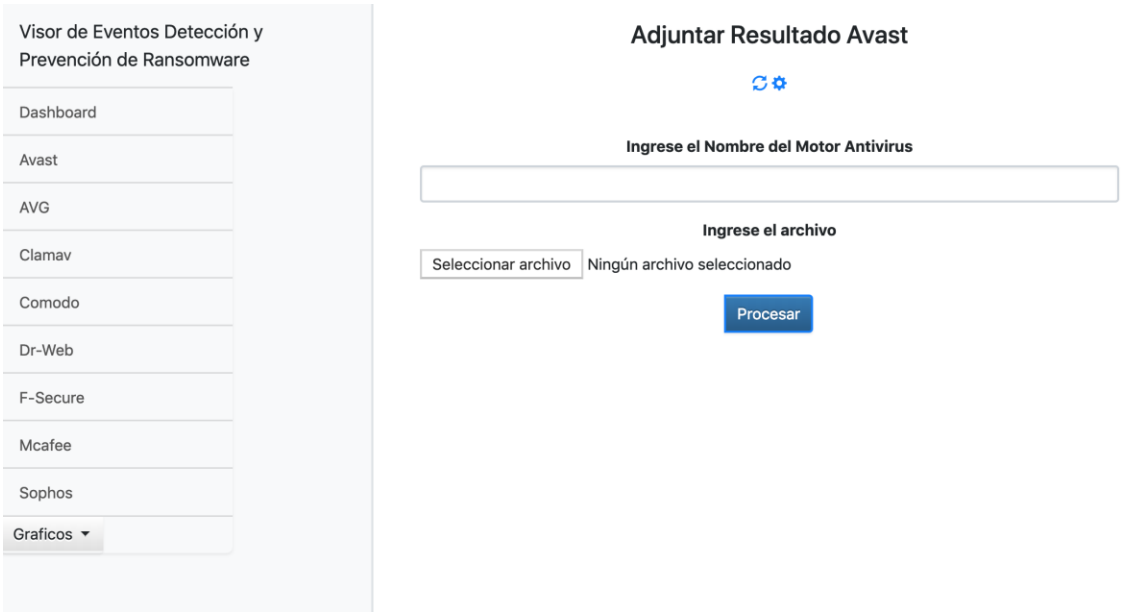


Id	Motor Antivirus	Infectado	Tipo de Ransomware	Engine	Actualizacion	Fecha de Analisis
2	Mcafee	Si	Ransom-O	5600.1067	20180928	2019-03-06 11:15:00
3	Mcafee	Si	Ransom-O	5600.1067	20180928	2019-03-06 11:15:00
4	Mcafee	Si	Ransom-O	5600.1067	20180928	2019-02-27 11:15:00
6	AVG	Si	Trojan horse Generic_r.HMF	13.0.3114	20180814	2019-03-07 02:20:00
7	AVG	Si	Trojan horse SHeur4.CJHN	13.0.3114	20180814	2019-01-16 00:00:00
8	Clamav	No		0.100.1	20180928	2019-03-06 11:15:00
12	Mcafee	Si	Ransom-O	5600.1067	20180928	2019-03-10 17:06:11
13	Mcafee	Si	Ransom-O	5600.1067	20180928	2019-03-10 17:06:16

Fuente: Elaboración propia.

Para el ingreso de los resultados generados o exportados por Malice, utilizamos las opciones motoras de antivirus según sea el caso, para ello damos clic en cualquiera de las firmas de antivirus que aparecen en la herramienta, y aparece una vista como la que aparece en la Figura 4-9. Ingresamos el nombre del motor de firma, adjuntamos el archivo y damos clic en procesar, y automáticamente se almacena el registro.

Figura 4-10. Ingreso de Resultados de *Ransomware*.



Fuente: Elaboración propia.

Para la visualización de los resultados damos clic en la opción Dashboard dentro de la aplicación, ver Figura 4-11, y en esta opción se visualizan todos los resultados ingresados en la herramienta.

Figura 4-11. Dashboard de resultados de detección y prevención de *ransomware*.

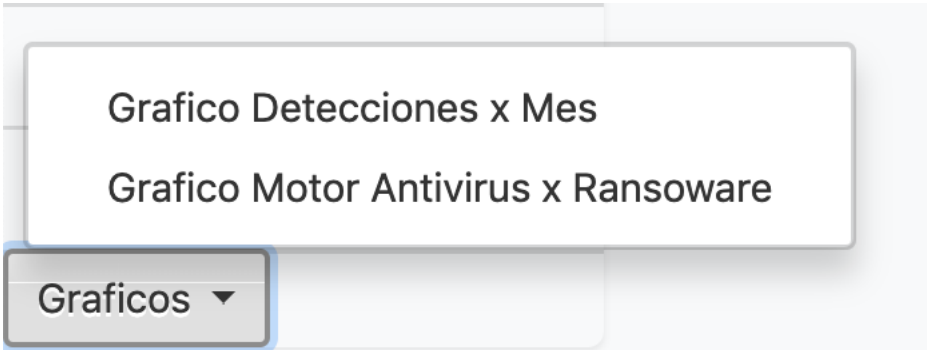
Resultados de Detección y Prevención

Id	Motor Antivirus	Infectado	Tipo de Ransomware	Engine	Actualizacion	Fecha de Analisis
2	Mcafee	Si	Ransom-O	5600.1067	20180928	2019-03-06 11:15:00
3	Mcafee	Si	Ransom-O	5600.1067	20180928	2019-03-06 11:15:00
4	Mcafee	Si	Ransom-O	5600.1067	20180928	2019-02-27 11:15:00
6	AVG	Si	Trojan horse Generic_r.HMF	13.0.3114	20180814	2019-03-07 02:20:00
7	AVG	Si	Trojan horse SHeur4.CJHN	13.0.3114	20180814	2019-01-16 00:00:00
8	Clamav	No		0.100.1	20180928	2019-03-06 11:15:00
12	Mcafee	Si	Ransom-O	5600.1067	20180928	2019-03-10 17:06:11
13	Mcafee	Si	Ransom-O	5600.1067	20180928	2019-03-10 17:06:16

Fuente: Elaboración propia.

Para la visualización de los gráficos tenemos la opción Gráficos, donde aparecen dos opciones, la primera opción es Gráfico Detección x Mes; la segunda opción es Gráfico Motor Antivirus x *Ransomware*, ver Figura 4-12.

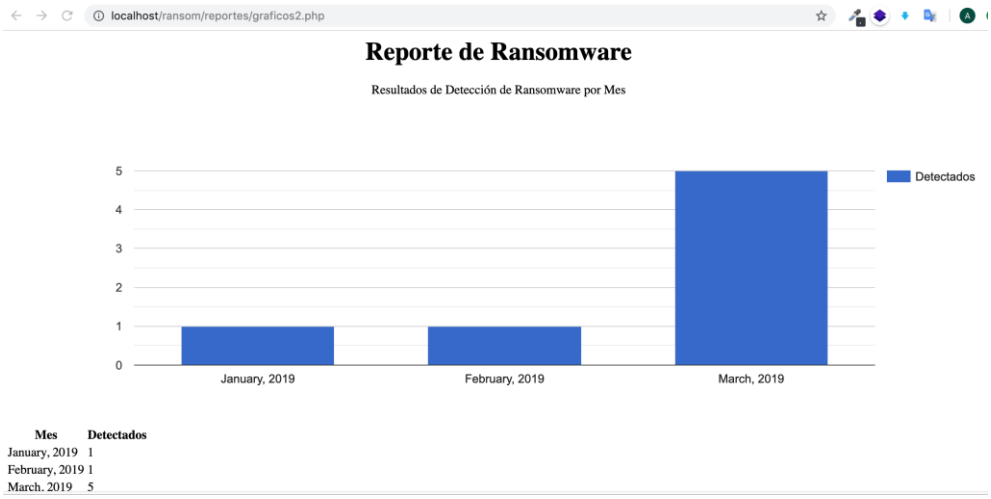
Figura 4-12. Gráficos de Herramienta de Detección y Prevención de *Ransomware*.



Fuente: Elaboración propia.

En la primera opción de Gráfico Detección x Mes, se pueden visualizar todas las detecciones en cada uno de los meses en los cuales se ha realizado la carga de archivos generados por Malice, ver Figura 4-13.

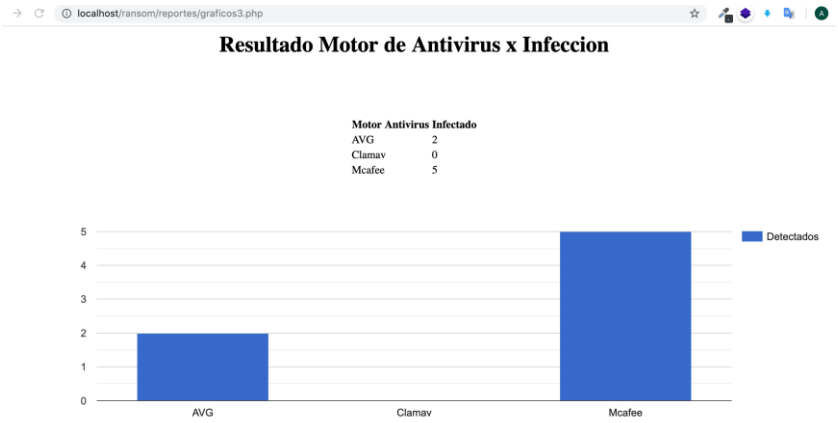
Figura 4-13. Reporte de detecciones x mes.



Fuente: Elaboración propia.

En la segunda opción Gráfico Motor Antivirus x *Ransomware,* se visualizan las estadísticas de cada firma de antivirus con la cantidad de detecciones, esto con el fin de saber qué firmas tienen mayor índice de detección y prevención de *ransomware*, ver Figura 4-14.

Figura 4-14. Resultados Motor Antivirus x Infección



Fuente: Elaboración propia.

4.4 VALIDACIÓN DEL ESQUEMA METODOLÓGICO PARA DETECCIÓN Y PREVENCIÓN DE RANSOMWARE

Este capítulo tiene como objeto mostrar la validación del esquema metodológico mediante la evaluación de los métodos de detección y prevención frente a cada una de las variantes y lograr el último objetivo de este trabajo de grado. Con el fin de validar los resultados se debe analizar el Anexo 5-1. Validación del Esquema Metodológico con Métodos de Detección y Prevención de *Ransomware* para una mayor comprensión de este capítulo.

4.4.1 Validación de Métodos de Prevención y Detección

Con el fin de crear un instrumento de validación, se aplicaron 40 métodos de detección y prevención de la sección 1.2, frente a las 8 variantes de *ransomware* seleccionadas como prueba piloto. Este muestreo ha sido básico para la determinación del tamaño muestral, aunque previamente se definieron las siguientes características: confiabilidad del 95% y error del 5%.

La fórmula para determinar la sumatoria de los métodos de prevención es la siguiente:

VTMP = $\sum MP$

Donde

VTP = Valoración Total Métodos de Prevención

$\sum MP$ = Sumatoria de Métodos de Prevención

Donde la sumatoria por método de prevención frente a cada una de las variantes de *ransomware* fue de 40, que correspondió al total de métodos, debido a que, métodos de prevención puede ser cualquier control que evite la ejecución de un *ransomware* y representa el grueso de la muestra con el 100%.

La fórmula para determinar la sumatoria de los métodos de detección es la siguiente:

VTD = $\sum MD$ / Total de variantes

Donde

VTD = Valoración Total Métodos de Detección

$\sum MD$ = Sumatoria de Métodos de Detección

Donde la sumatoria por método de detección, frente a cada una de las variantes de *ransomware* fue de 12 métodos que pueden detectar un *ransomware* en una estación de trabajo, esto debido a que, los métodos no tienen la capacidad de bloquear o impedir la ejecución de un *ransomware* y representa una pequeña porción de la muestra con más del 30%.

4.4.2 Porcentaje de Validación Métodos de Prevención y Detección

Porcentaje de prevención.

Para el cálculo del porcentaje de prevención utilizamos la siguiente la fórmula:

P_{TMD} = $\sum MP$ / Total de Métodos de Prevención x 100

Donde

P_{TMP} = Porcentaje Total Métodos de Prevención

$\sum MD$ = Sumatoria de Métodos de Prevención

T_m = Total de métodos

$P_{TMP}= \frac{40}{40} x\ 100$

$P_{TMP}= 100\%$

Para el cálculo del porcentaje de detección.

La fórmula para determinar el porcentaje de los métodos de detección es la siguiente:

P_{TMD} = $\sum MD$ / Total de Métodos de Detección x 100

Donde

P_{TMD} = Porcentaje Total Métodos de Detección

$\sum MD$ = Sumatoria de Métodos de Detección

T_m = Total de métodos

$P_{TMD}= \frac{12}{40} x\ 100$

$P_{TMD}= 30\%$

4.4.3 Resultados consolidados

Los resultados de este trabajo de investigación se componen mediante cada una las fases metodológicas, donde en cada una de ellas se presentan los anexos y documentos necesarios que soportan y dan validez a esta investigación. Los resultados para la caracterización y clasificación de cada una de las variantes de *ransomware* representados en los anexos que sustentan esta investigación. Una vez realizada la caracterización de las variantes, se comprendió el funcionamiento y el comportamiento cada una de estas, donde se identificaron los patrones más comunes y sustraer los IoC de las variantes seleccionadas para posteriormente realizar un esquema mediante los métodos o controles más adecuados de detección y prevención.

Mediante los conceptos identificados de acuerdo con el conjunto de investigaciones de autores y entidades del campo de la ciberseguridad, se logró identificar las vulnerabilidades, debilidades y los riesgos de cada una de las variantes de *ransomware* seleccionadas, para posteriormente enunciar los 40 controles o métodos de detección y prevención como medidas de seguridad necesarias para aplicarlos en cualquier tipo de organización de manera exitosa. Es necesario aplicar los controles basados en el esquema metodológico para la detección y prevención de *ransomware* para así mitigar las amenazas generadas por este en cualquier tipo organización debido a que el 100% de los controles de prevención son efectivos.

En las pruebas realizadas a las diferentes variantes de *ransomware* y aplicando técnicas forenses de análisis estático y dinámico en un ambiente controlados se confirma la validez y aplicabilidad del esquema metodológico y del desarrollo de la herramienta para la detección y prevención de *ransomware*. Dicha herramienta identifica y visualiza si una variante de *ransomware* es detectada mediante Malice, Yara y con cada firma de antivirus, para luego visualizar el contenido del resultado en la herramienta desarrollada en un ambiente web.

Cabe mencionar que el porcentaje de detección y prevención mediante Malice es del 43% del conjunto de las variantes de *ransomware* analizadas y seleccionadas. Donde la ejecución de la herramienta Malice puede presentar riesgos si no se tienen en cuenta los controles de seguridad necesarios.

5. Conclusiones y Recomendaciones

5.1 Conclusiones

Gracias a la labor de este trabajo de grado se presentaron los comportamientos y el funcionamiento de algunas muestras de *ransomware*, además de los problemas que se generan de tener métodos independientes para la detección y prevención de *ransomware*, debido al no estar de forma estructurada, organizada y simplificada, además se presentó el ciclo de vida del *ransomware* y los vectores de ataque donde a partir de ahí se construyó un esquema metodológico para la detección y prevención contra *ransomware* mediante los métodos existentes tanto teóricos como prácticos. También se realizó el desarrollo de una herramienta para la prevención de ransomware y la validación del mismo. De manera general estas fueron las conclusiones del trabajo de grado.

Mediante la caracterización de las 22 muestras y variantes de *ransomware* se logró identificar el tipo de familia, patrones y comportamientos más utilizados, algoritmos de cifrado, vectores de infección y propagación, métodos de pago y cantidad de pago. La importancia de estudiar la forma de operación del *ransomware* radica en su comportamiento para posteriormente tomar medidas de mitigación y protección más efectivas a medida que este va evolucionando.

Un esquema de metodológico para la detección de *ransomware* en una estación de trabajo permitió tener medidas de control más específicas durante el ciclo de vida de detección y prevención de un *ransomware,* ayudando a los administradores de TI o especialistas de seguridad a tener un control más detallado sobre los equipos o estaciones de trabajo, además dando mayor visibilidad a los riesgos generados por las variantes de *ransomware* y minimizando estos riesgos mediante la herramienta de prevención y detección de *ransomware.*

Mediante el desarrollo de la herramienta propuesta, es importante considerar que si bien el porcentaje de efectividad total no es el adecuado, debido a que el conjunto total de firmas en su gran mayoría no realiza una detección y prevención de *ransomware*, solo es posible lograr un mayor porcentaje de efectividad si se realiza un análisis de las firmas de antivirus como virus total, sophos, avg mcafee ya que tienen el 90% de efectividad.

Los resultados de validación muestran que el esquema metodológico mediante los 40 métodos de detección y prevención propuestos en esta tesis, pueden ser utilizados de forma exitosa en una estacion de trabajo con sistema operativo Windows, comprobando que tienen un grado de efectividad del 100% para los métodos de prevención y un 27% para los métodos de detección.

.

5.2 Recomendaciones

Gracias a los hallazgos obtenidos en el presente trabajo, a nivel técnico, se sugiere realizar un ejercicio similar a escala mayor, con el fin de identificar y generar nuevos esquemas metodológicos para la prevención y detección de *ransomware* y que puedan ser utilizados como base para entender y describir nuevas variaciones del programa maligno tipo *ransomware*, enfocadas para las nuevas tecnologías como son los dispositivos móviles, Smart TV, CCTV, dispositivos IoT, sistemas SCADA entre otros. Donde se deben tener otros métodos de detección y prevención debido a la forma como se comporta cada variante de *ransomware* en cada uno de estos elementos varían debido a que tienen diferentes arquitecturas y sistemas operativos distintas cambiando el ciclo de vida del *ransomware* y por ende el ciclo de vida de detección y prevención de un *ransomware*. En los trabajos futuros podría analizarse la posibilidad de presentar un esquema metodológico para la detección y prevención de *ransomware* para todas las plataformas, dispositivos móviles y Smart tv.

Adicionalmente, las técnicas descritas en esta tesis se basan en el sistema operativo Windows. Sin embargo, se utilizan técnicas y métodos de detección y prevención similares para otras plataformas como Android, Linux y Mac, así que podría haber un aumento en las técnicas de infección en dichos S.O en algún momento.

A. Anexo:

Anexo 2-1. Familias de Ransomware.docx

Anexo 2-2. Ransomware Comportamientos.xlsx

Anexo 3-1. Análisis Comportamientos de Ransomware.xlsx

Anexo 3-2. Herramientas de Recuperación Ransomware.xlsx

Anexo 4-1. Herramientas de Análisis de Malware.xlsx

Anexo 4-2. Indicadores de Compromiso Ransomware

Anexo 4-3. Instalación Malice.docx

Anexo 4-4. Instalación Python y Yara.docx

Anexo 4-5. Evaluación de Herramienta Ransomware.xlsx

Anexo 4-6 Scripts de Ejecución

Anexo 4-7 Resultados

Anexo 5-1. Validación del Esquema Metodológico con Métodos de Detección y Prevención de Ransomware.xlsx

Bibliografía

Abhijit Mohanta, Mounir Hahad, Kumaraguru Velmurugan. (2018, octubre 3). Preventing Ransomware. Recuperado 20 de enero de 2019, de https://www.packtpub.com/application-development/preventing-ransomware

Adamov, A., & Carlsson, A. (2017). The state of ransomware. Trends and mitigation techniques. *2017 IEEE East-West Design Test Symposium (EWDTS)*, 1-8. https://doi.org/10.1109/EWDTS.2017.8110056

Anton Chuvakin. (2017, febrero 5). On UEBA / UBA Use Cases. Recuperado 27 de junio de 2018, de https://blogs.gartner.com/anton-chuvakin/2017/01/05/on-ueba-uba-use-cases/

Aziz, S. (2016a). Ransomware in High-Risk Environments. *Information Technology Capstone Research Project Reports*. Recuperado de https://scholar.valpo.edu/itcrpr/1

Aziz, S. (2016b). Ransomware in High-Risk Environments. *Information Technology Capstone Research Project Reports*. Recuperado de http://scholar.valpo.edu/itcrpr/1

Bhardwaj, A., Subrahmanyam, G. V. B., Avasthi, V., & Sastry, H. (2015). Ransomware: A Rising Threat of new age Digital Extortion. *arXiv:1512.01980 [cs]*. Recuperado de http://arxiv.org/abs/1512.01980

Brewer, R. (2016). Ransomware attacks: detection, prevention and cure. *Network Security*, *2016*(9), 5-9. https://doi.org/10.1016/S1353-4858(16)30086-1

Bridges, L. (2008). The changing face of malware. *Network Security*, *2008*(1), 17-20. https://doi.org/10.1016/S1353-4858(08)70010-2

Cabaj, K., & Mazurczyk, W. (2016). Using Software-Defined Networking for Ransomware Mitigation: The Case of CryptoWall. *IEEE Network*, *30*(6), 14-20. https://doi.org/10.1109/MNET.2016.1600110NM

Caivano, D., Canfora, G., Cocomazzi, A., Pirozzi, A., & Visaggio, C. A. (2017). Ransomware at X-Rays. *2017 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, 348-353. https://doi.org/10.1109/iThings-

GreenCom-CPSCom-SmartData.2017.58

Christopher M Frenz, & Christian Diaz. (2018, diciembre 3). OWASP Anti-Ransomware Guide Project - OWASP. Recuperado 11 de octubre de 2018, de https://www.owasp.org/index.php/OWASP_Anti-Ransomware_Guide_Project

Darragh Delaney. (2016, mayo 16). 5 Methods For Detecting Ransomware Activity. Recuperado 7 de agosto de 2018, de https://www.netfort.com/blog/methods-for-detecting-ransomware-activity/

Denis Nazarov, Olga Emelyanova. (2006, julio 6). Blackmailer: the story of Gpcode. Recuperado de Securelist - Kaspersky Lab's cyberthreat research and reports website: https://securelist.com/blackmailer-the-story-of-gpcode/36089/

Duc, H. N. (2016, febrero 11). YARA is a tool aimed at identify and classify malware samples - interview with creator Victor M. Alvarez. Recuperado 24 de junio de 2018, de Pentestmag website: https://pentestmag.com/yara-is-a-tool-aimed-at-identify-and-classify-malware-samples-interview-with-creator-victor-m-alvarez/

Eset. (2016, septiembre 9). Definición de virus, códigos maliciosos y ataques remotos—Base de conocimiento ESET. Recuperado 11 de noviembre de 2017, de https://soporte.eset-la.com/kb186/?locale=es_ES

first.org. (2019, junio). CVSS v3.1 Specification Document. Recuperado 17 de junio de 2019, de FIRST — Forum of Incident Response and Security Teams website: https://www.first.org/cvss/specification-document

Gaviria Pablo. (2016). *Aplicación de Metodología de Malware para el Análisis de la amenaza avanzada persistente (APT) "Poison Ivy*.

Gazet, A. (2010). Comparative analysis of various ransomware virii. *Journal in Computer Virology*, *6*(1), 77-90. https://doi.org/10.1007/s11416-008-0092-2

Gonzalez, D., & Hayajneh, T. (2017). Detection and prevention of crypto-ransomware. *2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON)*, 472-478. https://doi.org/10.1109/UEMCON.2017.8249052

Grance, T., Kent, K., & Kim, B. (2004). Computer Security Incident Handling Guide. *Special Publication (NIST SP) - 800-61*. Recuperado de https://www.nist.gov/publications/computer-security-incident-handling-guide-1

Hosmer, C., Bartolomie, J., & Pelli, R. (2016). Chapter 1 - The Impact of Windows Command Line Investigations. En *Executing Windows Command Line Investigations* (pp. 1-9). https://doi.org/10.1016/B978-0-12-809268-2.00001-8

IBM. (2016, diciembre 14). IBM News room - 2016-12-14 IBM Study: Businesses More likely to Pay Ransomware than Consumers - United States. Recuperado 14 de abril de 2017, de https://www-03.ibm.com/press/us/en/pressrelease/51230.wss

J. A. Gómez-Hernández, Pedro García-Teodoro, & L. Álvarez-González. (2018). R-Locker: Thwarting ransomware action through a honeyfile-based approach. *Computers & Security*, *73*, 389-398. https://doi.org/10.1016/j.cose.2017.11.019

Jatinder N.D. Gupta, S. K. S. (2008). *Handbook of Research on Information Security and Assurance*.

Jeromie Jackson. (2016, mayo 18). User Behavior Analytics (UBA) & Ransomware Analytics. Recuperado 29 de junio de 2018, de Virtual CISO website: https://itknowledgeexchange.techtarget.com/virtual-ciso/user-behavior-analytics-uba-ransomware-analytics/

Kao, C. H., Chi, P., & Lee, Y. (2014). Automatic Testing Framework for Virtualization Environment. *2014 IEEE International Symposium on Software Reliability Engineering Workshops*, 134-135. https://doi.org/10.1109/ISSREW.2014.28

Kaspersky Lab. (2016). *Kaspersky story of the year ransomware revolution*.

Kevin Savage, Peter Coogan, & Hon Lau. (2015). *The evolution of ransomware*.

Kharraz, A., Robertson, W., Balzarotti, D., Bilge, L., & Kirda, E. (2015). Cutting the Gordian Knot: A Look Under the Hood of Ransomware Attacks. *Detection of Intrusions and Malware, and Vulnerability Assessment*, 3-24. https://doi.org/10.1007/978-3-319-20550-2_1

Krunal, G., & Viral, P. (2017). Survey on Ransomware: A New Era of Cyber Attack. *International Journal of Computer Applications*, *168*, 38-41. https://doi.org/10.5120/ijca2017914446

Liu, S., & Kuhn, R. (2010). Data Loss Prevention. *IT Professional*, *12*(2), 10-13. https://doi.org/10.1109/MITP.2010.52

Manuel Zaforas. (2017, noviembre 13). ¿Es Python el lenguaje del futuro? - Paradigma. Recuperado 26 de septiembre de 2018, de https://www.paradigmadigital.com/dev/es-python-el-lenguaje-del-futuro/

Mark Nicolett, & Amrit T. Williams. (2005, febrero 5). Improve IT Security With Vulnerability Management. Recuperado 7 de agosto de 2018, de https://www.gartner.com/doc/480703/improve-it-security-vulnerability-management

Mell, P., Kent, K., & Nusbaum, J. (2005). *Guide to Malware Incident Prevention and Handling* (N.º NIST Special Publication (SP) 800-83 (Withdrawn)). https://doi.org/10.6028/NIST.SP.800-83

Melo, G. E. R. (2011). Apropiación y masificación de las tecnologías de la información y las comunicaciones (TIC) en las Mipyme. *Criterio Libre*, (15), 18.

Monika, Zavarsky, P., & Lindskog, D. (2016). Experimental Analysis of Ransomware on Windows and Android Platforms: Evolution and Characterization. *Procedia Computer Science*, *94*, 465-472. https://doi.org/10.1016/j.procs.2016.08.072

Moore, C. (2016). Detecting Ransomware with Honeypot Techniques. *2016 Cybersecurity and Cyberforensics Conference (CCC)*, 77-81. https://doi.org/10.1109/CCC.2016.14

Nate Lord. (2017, julio 7). What are Indicators of Compromise? [Text]. Recuperado 7 de agosto de 2018, de Digital Guardian website: https://digitalguardian.com/blog/what-are-indicators-compromise

Nath, H. V., & Mehtre, B. M. (2014). Static Malware Analysis Using Machine Learning Methods. *Recent Trends in Computer Networks and Distributed Systems Security*, 440-450. https://doi.org/10.1007/978-3-642-54525-2_39

Nieuwenhuizen, D. (2017). *A behavioural-based approach to ransomware detection*. 20.

P Tailor, J., & Patel, A. (2017). A Comprehensive Survey: Ransomware Attacks Prevention, Monitoring and Damage Control. *International Journal of Scientific Research*, *4*.

Philip OKane ; Sakir Sezer;Kieran McLaughlin. (2011, enero 10). Obfuscation: The Hidden Malware - IEEE Journals & Magazine. Recuperado de https://ieeexplore.ieee.org/abstract/document/5975134

Preneel, B. (2010, diciembre 12). *Cryptographic Hash Functions: Theory and Practice*. *6498*, 1-3. https://doi.org/10.1007/978-3-642-17401-8_9

Rad, B. B., Masrom, M., & Ibrahim, S. (2012). *Camouflage in Malware: from Encryption to Metamorphism*. 10.

Rebecca Wilson. (2017, septiembre 22). 74% of security incidents come from within organisations,

according to research - Recruitment International. Recuperado 11 de octubre de 2018, de https://www.recruitment-international.co.uk/blog/2017/09/74-percent-of-security-incidents-come-from-within-organisations-according-to-research

Salvi, Miss. H., & Kerkar, M. R. V. (2017). Ransomware: A Cyber Extortion. *ASIAN JOURNAL FOR CONVERGENCE IN TECHNOLOGY (AJCT) -UGC LISTED*, *2*(2). https://doi.org/10.1212/ajct.v2i2.55

Sanatinia, A., & Noubir, G. (2015). OnionBots: Subverting Privacy Infrastructure for Cyber Attacks. *2015 45th Annual IEEE/IFIP International Conference on Dependable Systems and Networks*, 69-80. https://doi.org/10.1109/DSN.2015.40

Segu-Info. (2016, marzo 31). Cómo evitar infectarse con archivos JS adjuntos y ransomware ~ Segu-Info. Recuperado 11 de octubre de 2018, de https://blog.segu-info.com.ar/2016/03/como-evitar-infectarse-con-archivos-js.html

Sgandurra, D., Muñoz-González, L., Mohsen, R., & Lupu, E. C. (2016). Automated Dynamic Analysis of Ransomware: Benefits, Limitations and use for Detection. *arXiv:1609.03020 [cs]*. Recuperado de http://arxiv.org/abs/1609.03020

Smith, G. (2016, agosto 29). History of Ransomware – The Never-Ending Threat | CFOC.ORG. Recuperado 2 de mayo de 2018, de https://cfoc.org/history-of-ransomware-the-never-ending-threat/

Spiceworks. (2016, mayo 13). Prevent ransomware by using FSRM. Recuperado 3 de agosto de 2018, de https://community.spiceworks.com/how_to/128744-prevent-ransomware-by-using-fsrm

Symantec. (2018, febrero 3). Email Gateway Security - Messaging Gateway | Symantec. Recuperado 11 de octubre de 2018, de https://www.symantec.com/products/messaging-gateway

Tim Buntrock. (2016, noviembre 4). How to Block Viruses and Ransomware Using Software Restriction Policies. Recuperado 3 de agosto de 2018, de Windows OS Hub website: http://woshub.com/how-to-block-viruses-and-ransomware-using-software-restriction-policies/

Trabelsi, Z., & Molvizadah, V. (2016). Edu-firewall device: An advanced firewall hardware device for information security education. *2016 13th IEEE Annual Consumer Communications Networking Conference (CCNC)*, 278-279. https://doi.org/10.1109/CCNC.2016.7444779

TrendMicro. (2011, enero 12). SMS Ransomware Tricks Russian Users - TrendLabs Security Intelligence Blog. Recuperado 9 de septiembre de 2018, de https://blog.trendmicro.com/trendlabs-security-intelligence/sms-ransomware-tricks-russian-users/

TrendMicro. (2016). Ransomware - Definition - Trend Micro USA. Recuperado 13 de noviembre de 2017, de https://www.trendmicro.com/vinfo/us/security/definition/ransomware

Tripwire. (2016, enero 24). 22 Ransomware Prevention Tips. Recuperado 7 de agosto de 2018, de The State of Security website: https://www.tripwire.com/state-of-security/security-data-protection/cyber-security/22-ransomware-prevention-tips/

Trisha. (2015, diciembre 18). OfficeMalScanner : Scan Office Documents for Macros Before Opening. Recuperado 7 de agosto de 2018, de TrishTech.com website: https://www.trishtech.com/2015/12/officemalscanner-scan-office-documents-for-macros-before-opening/

Vadim Kotov; Fabio Massacci. (2013, enero 10). Anatomy of Exploit Kits | SpringerLink. Recuperado 21 de enero de 2019, de https://link.springer.com/chapter/10.1007/978-3-642-36563-8_13

Vanderburg, E. (2017, agosto 29). The evolution of a cybercrime: A timeline of ransomware advances. Recuperado 2 de mayo de 2018, de https://www.linkedin.com/pulse/evolution-cybercrime-timeline-ransomware-advances-eric-vanderburg

Vasanthi, S., & Chandrasekar, S. (2011). A study on network intrusion detection and prevention system current status and challenging issues. *3rd International Conference on Advances in Recent Technologies in Communication and Computing (ARTCom 2011)*, 181-183. https://doi.org/10.1049/ic.2011.0075

Xiyang, Z., & Chuanqing, C. (2009). Research on VLAN Technology in L3 Switch. *2009 Third International Symposium on Intelligent Information Technology Application*, *3*, 722-725. https://doi.org/10.1109/IITA.2009.498

Young, A., & Yung, M. (1996). *Cryptovirology: Extortion-Based Security Threats and Countermeasures*.

Zorabedian, J. (2015, marzo 3). Anatomy of a ransomware attack: CryptoLocker, CryptoWall, and how to stay safe (Infographic). Recuperado 18 de abril de 2018, de Sophos News website: https://news.sophos.com/en-us/2015/03/03/anatomy-of-a-ransomware-attack-cryptolocker-cryptowall-and-how-to-stay-safe-infographic/