# UNIVERSIDAD POLITÉCNICA SALESIANA
## SEDE CUENCA

## CARRERA DE INGENIERÍA DE SISTEMAS

*Trabajo de titulación previo*
*a la obtención del título de*
*Ingeniero de Sistemas*

**PROYECTO TÉCNICO:**

**"PROTOTIPO DE SOFTWARE DE ADMINISTRACIÓN REMOTA"**


**AUTORES:**

ANDREA CAROLINA CALDERON OJEDA
GEOVANNY PAUL MOROCHO MENDEZ

**TUTOR:**

PhD. PABLO LEONIDAS GALLEGOS SEGOVIA



CUENCA - ECUADOR

2020

# CESIÓN DE DERECHOS DE AUTOR

Nosotros, Andrea Carolina Calderon Ojeda con documento de identificación N˚ 070662222-2 y Geovanny Paul Morocho Mendez con documento de identificación N˚ 010665706-7, manifestamos nuestra voluntad y cedemos a la Universidad Politécnica Salesiana, la titularidad sobre los derechos patrimoniales en virtud de que somos autores del trabajo de titulación: **"PROTOTIPO DE SOFTWARE DE ADMINISTRACIÓN REMOTA"**, mismo que ha sido desarrollado para optar por el título de: *Ingeniero de Sistemas*, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En aplicación a lo determinado en la Ley de Propiedad Intelectual, en nuestra condición de autores, nos reservamos los derechos morales de la obra antes citada. En concordancia, suscribimos este documento en el momento que hacemos entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Cuenca, febrero del 2020

Andrea Carolina Calderon Ojeda
C.I. 070662222-2

Geovanny Paul Morocho Mendez
C.I. 010665706-7

# CERTIFICACIÓN

Yo, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: **"PROTOTIPO DE SOFTWARE DE ADMINISTRACIÓN REMOTA",** realizado por Andrea Carolina Calderon Ojeda y Geovanny Paul Morocho Mendez, obteniendo el *Proyecto Técnico*, que cumple con todos los requisitos estipulados por la Universidad Politécnica Salesiana.

Cuenca, febrero del 2020

Dr. Pablo Leonidas Gallegos Segovia
CI: 010259358-9

# DECLARATORIA DE RESPONSABILIDAD

Nosotros, Andrea Carolina Calderon Ojeda con documento de identificación N° 0706622222 y Geovanny Paul Morocho Mendez con documento de identificación N° 0106657067, autores del trabajo de titulación: **"PROTOTIPO DE SOFTWARE DE ADMINISTRACIÓN REMOTA",** certificamos que el total contenido de este *Proyecto Técnico* es de nuestra exclusiva responsabilidad y autoría.

Cuenca, febrero del 2020

Andrea Carolina Calderon Ojeda
C.I. 070662222-2

Geovanny Paul Morocho Mendez
C.I. 010665706-7

# AGRADECIMIENTOS

Primeramente, agradecemos a Dios por brindarnos salud y colmarnos de bendiciones para poder culminar con nuestra carrera universitaria, por ponernos a lo largo de este camino a las personas indicadas que nos han acompañado siempre. A nuestra familia de manera especial a nuestros padres y hermanos por todo el amor, confianza, dedicación, esfuerzo y ese apoyo incondicional para no dejarnos desfallecer en todos estos años de formación académica, gracias por motivarnos a ser mejores personas cada día. A nuestros docentes por compartir con nosotros sus conocimientos, especialmente a nuestro tutor de tesis PhD. Pablo Gallegos por su apoyo y dedicación para la culminación de nuestro proyecto.

Gracias a todos.

Andrea Carolina Calderon Ojeda.
Geovanny Paul Morocho Mendez.

# DEDICATORIA

Esta tesis la quiero dedicar a mi familia por siempre apoyarme en cada decisión que tomo y por creer en mí. De manera especial a mis padres María y Tito por su sacrificio y esfuerzo, por ser mis modelos a seguir, por ser mi soporte, mi compañía y esas fuerzas constantes acompañadas de consejos cuando eh estado en situaciones difíciles. A mi hermano Josue que por ser quien me impulsa a ser mejor persona cada día y su apoyo incondicional. A mi tío Arturo que ha estado presente en toda mi vida y siempre me ha brindado su confianza. A todos aquellos que fueron parte de esta formación académica de los cuales me llevo gratos momentos.

Andrea Carolina Calderon Ojeda.

Esta tesis la dedico con todo mi cariño y afecto a mis padres, por su sacrificio y esfuerzo en todos estos años, por estar presentes no solo en esta etapa tan importante de mi vida, sino en todo momento ofreciéndome su consejo y apoyo para ser mejor persona y un excelente profesional. Gracias a mi Mamá por su apoyo y por ser el motor principal para cumplir mis sueños, gracias por confiar y creer en mí y en mis expectativas, a mi Papá por cada consejo y deseo de buscar lo mejor para mi vida. A mis hermanos por estar siempre presentes brindándome su apoyo incondicional en los momentos que más los necesité, a mis compañeros y amigos con los cuales compartimos momentos inolvidables en cada una de las clases.

Geovanny Paul Morocho Mendez.

# Tabla de contenido

# Resumen

En la actualidad, los activos de la seguridad de la información se han convertido en el valor más preciado de las empresas y las personas, sin embargo, nuevas amenazas están presentes en la red de computadoras, por lo que las tecnologías de la información han dejado atrás los perímetros tradicionales y han evolucionado creando nuevos sistemas de seguridad que intentan minimizar el impacto de virus, Malware y Ransomware en las empresas. Al mismo tiempo los dispositivos móviles, el acceso no controlado a Internet por intervención del eslabón más débil de la cadena de seguridad "el usuario" crean un perímetro no controlado que permite a los atacantes llegar hasta la información sensible de las empresas. Para esto los atacantes usan técnicas de ingeniería social como la afinidad o coerción para convencer a los usuarios de descargar software malicioso que les permita infectar a los equipos de los usuarios y dispersarse por la red de la empresa. Los atacantes dentro de las técnicas pueden infectar sitios web o enviar correos electrónicos con códigos malignos con los denominados caballos de troya, estos programas se instalan en el computador y permite el acceso remoto del atacante, entre estas técnicas encontramos herramientas de acceso remoto (R.A.T) los cuales representan un ataque cibernético efectivo y eficiente. En la presente tesis se expone el desarrollo de un prototipo de software de administración remota dirigido a comprometer la seguridad de Windows en sus versiones 7 y 10 como herramienta de pruebas de contexto para asegurar el perímetro interno de la empresa ya que llega a la red interna evadiendo los Firewall y controles perimetrales. Nuestro prototipo de software está desarrollado en el lenguaje de programación C# en el entorno Visual Studio el cual proporciona las herramientas y el acceso al paquete de librerías y Forms propios de Windows. Sus funcionalidades son: administración de archivos, captura de los procesos que se están ejecutando en el administrador de tareas, ver información del sistema y escritorio remoto del equipo controlado. Por último, se demuestra el funcionamiento del software en tres posibles escenarios en donde aplicamos técnicas de ingeniería social y métodos de ofuscación para que el Malware pase inadvertido. En el primer escenario la víctima no cuenta con ningún mecanismo de seguridad, luego cuenta con protección de un Firewall y finalmente tiene protección del Firewall y antivirus.

# Abstract

Currently, information security assets have become the most precious value of companies and people, however, new threats are present in the computer network, which is why information technologies have left behind traditional perimeters and have evolved by creating new security systems that try to minimize the impact of viruses, Malware and Ransomware on businesses. At the same time, mobile devices, uncontrolled access to the Internet through the intervention of the weakest link in the security chain “the user” create an uncontrolled perimeter that allows attackers to reach sensitive company information.

For this, the attackers use social engineering techniques such as affinity or coercion to convince users to download malicious software that allows them to infect users' computers and disperse through the company's network. The attackers within the techniques can infect websites or send emails with malignant codes with the so-called trojan horses, these programs are installed on the computer and allow the attacker's remote access, among these techniques we find remote access tools (RAT) which represent an effective and efficient cyber-attack. This thesis describes the development of a prototype of remote administration software aimed at compromising the security of Windows in its versions 7 and 10 as a context testing tool to ensure the internal perimeter of the company as it reaches the internal network evading Firewalls and perimeter controls.

Our software prototype is developed in the programming language C # in the Visual Studio environment, which provides the tools and access to the Windows libraries and Forms package. Its functionalities are: file management, capture of the processes that are running in the task manager, view system information and remote desktop of the controlled computer. Finally, the operation of the software is demonstrated in three possible scenarios where we apply social engineering techniques and obfuscation methods so that Malware goes unnoticed. In the first scenario, the victim does not have any security mechanism, then has a Firewall protection and finally has Firewall and antivirus protection.

# 1. Introducción

En la actualidad en las telecomunicaciones los riesgos informáticos son mitigados a través de perímetros de seguridad como el uso de Firewall, antivirus, políticas etc. Sin embargo, estos no son suficientes para cubrir los ataques masivos a los que se enfrentan diariamente los sistemas informáticos. Desde hace muchos años atrás los hackers se vienen aprovechando del eslabón más débil de la cadena de seguridad "el usuario" optando diferentes herramientas para llegar a sus víctimas, predominando en la actualidad el Phishing como técnica de acceso a las pequeñas, medianas y grandes empresas e inclusive hasta los hogares. Por otro lado, en cuanto a la herramienta más utilizada para la abstracción y robo de información sobresalen los troyanos; estos son códigos maliciosos que se encuentra inmersos dentro de un archivo o contenido multimedia utilizando diferentes técnicas para llegar a la víctima. En esta tesis nos enfocamos en el desarrollo de un prototipo R.A.T (Troyano de Administración Remota) es de arquitectura Cliente-Servidor el mismo que permite controlar a un equipo independientemente del lugar geográfico en el que se encuentre. Finalmente, como sistema operativo objetivo seleccionamos Windows debido que a comparación con Linux y Mac Os es el sistema más atacado, por ende, nuestra herramienta esta direccionada a trabajar con las versiones más recientes, en las cuales exponemos escenarios donde vemos personas que no utilizan mecanismos de seguridad como; Firewall, antivirus y personas que si los utilizan. La finalidad del despliegue de estos escenarios es demostrar que un R.A.T. puede ser un enemigo altamente riesgoso dentro de la gestión de la seguridad de información de una empresa.

# 2. Objetivos

## 2.1. General

Desarrollar un prototipo de software de administración remota para el acceso fácil a datos, recursos y aplicaciones importantes de forma ubicua.

## 2.2. Especifico

- OE1. Elaborar el estado del arte sobre las bases de un servicio de acceso remoto y herramientas de desarrollo para la implementación del software.
- OE2. Diseñar modelo de desarrollo del software.
- OE3. Implementar plataforma de controlador de acceso remoto utilizando la metodología de RAT.
- OE4. Contrarrestar las limitaciones de un RAT para que funcione como servicio.
- OE5. Determinar métricas de configuraciones del software.
- OE6. Realizar pruebas de funcionamiento.

# 3. Marco Teórico

Dentro del proceso de elaboración del proyecto, hemos encontrado diferentes técnicas y metodología que utilizan los Malware para propagarse e infectar a la víctima. Es por ello que, para mayor comprensión de nuestro proyecto, se ha realizado un análisis literario de cada uno de estos elementos y temas relacionados con el mismo, los cuales se describen a continuación.

## 3.1. Seguridad de la Información

La seguridad de la información abarca un conjunto de procedimiento con los que cuenta cada empresa o institución, estos procedimientos de seguridad están enfocados en proteger la información privada y evitar que la misma pueda ser robada o alterada por algún agente externo.

### 3.1.1. Definición de Seguridad de la Información

Según nuestro análisis literario, el autor Soriano sostiene que la seguridad de la información no está limitada a la eliminación de virus, ni mucho menos a evitar que los hackers tengan acceso a la red. Afirma que la seguridad de la información abarca una serie de procedimientos que debe seguir una empresa con el fin de garantizar la protección de los datos confidenciales y los sistemas de información. También asegura que protege la información del acceso, uso, divulgación, alteración, modificación, lectura, inspección, registro o destrucción no autorizada. Por último, menciona que la seguridad de la información tiene que ver con la confidencialidad, integridad y disponibilidad de los datos, independientemente de su formato [1].

Por otra parte, Lemus afirma que la seguridad de la información tiene como fin la protección de los sistemas de la información del acceso, uso, divulgación, interrupción o destrucción no autorizada. Además, sostiene que es un conjunto de medidas preventivas

y reactivas de la organización y de los sistemas tecnológicos que permiten resguardar y proteger la información buscando mantener la confidencialidad, disponibilidad e integridad de la misma [2].

Así también, C. Laundon y P. Laundon definen que "la seguridad de la información es la prevención del acceso no autorizado, uso, divulgación, interrupción, modificación, lectura, inspección, registro o destrucción de la información y los sistemas de información [3]." Por otra parte, Escriva, Romero, Ramada y Pérez definen la seguridad de la información como un conjunto de medidas y procedimientos, tanto humanos como técnicos, que permiten proteger la integridad, confidencialidad y disponibilidad de la información, elementos que conforman la triada de seguridad de la información [4].

Finalmente, el Comité de Sistemas de Seguridad Nacional[1] (CNSS) define la seguridad de la información como la protección de la información y sus elementos críticos, incluidos los sistemas y el hardware que usan, almacenan y transmiten esa información [5]. En conclusión, todos los autores mencionados anteriormente convergen en la necesidad de la triada de la seguridad de la información, la misma que explicaremos a continuación.


*Ilustración 1 Triada de Seguridad de Información*

En la Ilustración 1, resaltamos los elementos que conforman el triángulo de seguridad de la información y a continuación detallaremos cada uno de ellos.

- **Disponibilidad:** hace referencia a que la información se encuentre disponible en el momento que se la requiera, que no se vea afectada por puntos de interrupción inoportuna o no autorizados.

- **Integridad:** se encarga que la información no sea cambiada por personas no autorizadas y así la información no sea mal obrada, es decir, garantiza que los sistemas de información se mantengan íntegros.

- **Confidencialidad:** otorga que solo las personas autorizadas puedan tener acceso a información detallada sobre un área relacionada, es decir que la información será abstraída solo por los puntos establecidos para que esta no sea divulgada.

Así mismo, se requieren otras aristas para la seguridad de la información como:

[1] https://www.cnss.gob.do/

- **Autenticación:** se encarga de verificar que una persona es quien dice ser, para lo cual esta debe pasar por un proceso en donde, por ejemplo; deberá iniciar sesión, en el cual tendrá que ingresar sus credenciales como usuario y contraseña o algo que lo identifique de los demás [6].

- **No Repudio:** este garantiza que la comunicación sea entablada y si existen fallas en la misma quede un registro, como un acuse de recibo, es decir prueba que los involucrados en una comunicación sean legítimos mediante firmas digitales.

  Como menciona Costas existen dos tipos de no repudio [7]:

  – **No repudio en origen:** el emisor no puede negar el envío. La prueba la crea el propio emisor y la recibe el destinatario.

  – **No repudio en destino:** el receptor no puede negar que recibió el mensaje porque el emisor tiene pruebas de la recepción. En este caso la prueba irrefutable la crea el receptor y la recibe el emisor.

### 3.1.2. Importancia de proteger la información

La información siempre ha sido el activo principal y más importante que posee una empresa, debido a que sin ella no se puede llevar un registro de todas sus actividades. Cuando se trata de información; hablamos de datos financieros, estrategias de mercado hasta los datos personales de cada individuo que esté ligado a la misma. La información genera conocimiento y es primordial para la toma de decisiones, permitiendo establecer una ventaja competitiva en el mercado. Es por ello que debe estar bien protegida y administrada, para evitar vulnerabilidades, debido a que siempre tiene la posibilidad de estar expuesta ante agentes externos que buscan apropiarse la información privada y usarlo para su propio beneficio [5], [8], [9].

Actualmente existe una metodología que permite el aseguramiento, confidencialidad e integridad de la información denominada SGSI[2], esta metodología se basa en el estándar internacional ISO 27001[3] el cual especifica todos los requisitos necesarios para una buena gestión de la información, integrando evaluaciones de vulnerabilidades y aplicación de controles para mitigar las mismas.

### 3.1.3. ¿Qué es él SGSI?

Según la Organización Internacional de Normalización[4] es un enfoque sistemático para administrar la información confidencial de la empresa para que permanezca segura. Esta incluye personas, procesos y sistemas de TI mediante la aplicación de un proceso de gestión de riesgos.

### 3.1.4. Fases de implementación de un SGSI

Para realizar una correcta implementación de este sistema debemos seguir ciertas fases que se muestran en la Ilustración 2 -adaptada de [10].

[2] Sistema de Gestión de la Seguridad de la Información.
[3] https://www.isotools.org/normas/riesgos-y-seguridad/iso-27001/
[4] https://www.iso.org/home.html

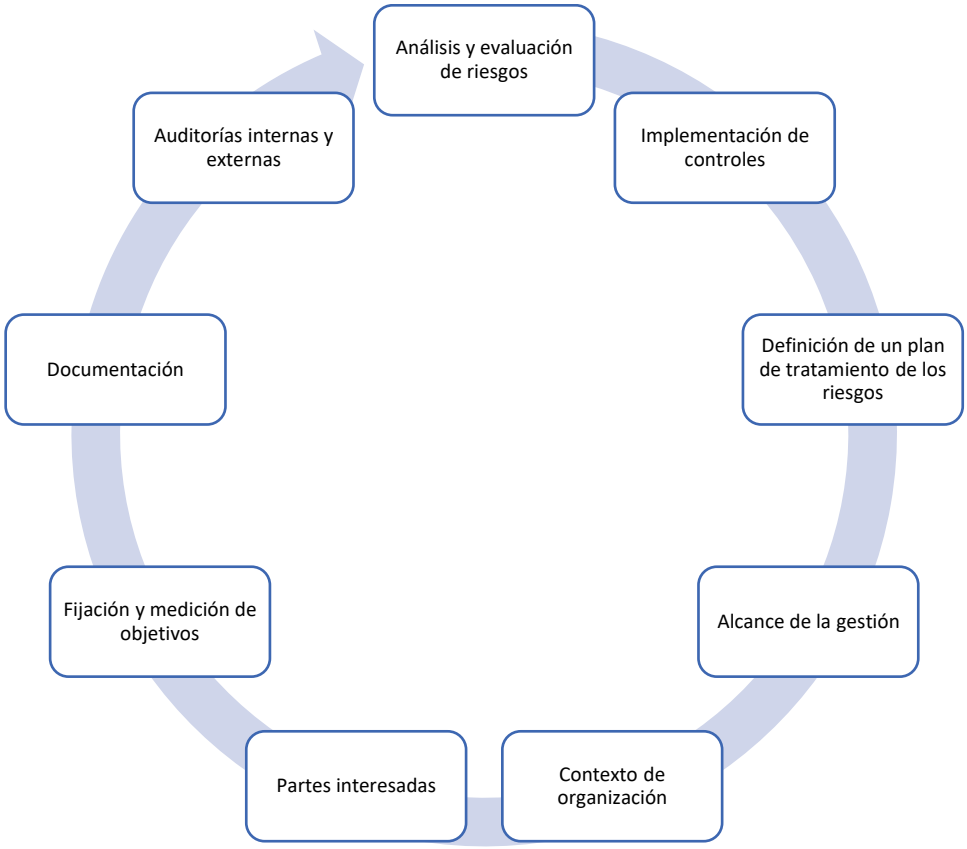


*Ilustración 2 Fases de implementación SGSI*

Describimos a continuación cual es el propósito de cada Fase.

Fase 1: Análisis y evaluación de riesgos

Dentro de esta etapa se toma en cuenta cada evento que afecta potencialmente a un activo de información este puede ser debido a; fallas técnicas, fallas a causa de eventos naturales o a causa de recursos humanos, etc. El objetivo es identificar los principales activos de la organización, determinar las vulnerabilidades y desarrollar un plan de gestión de riesgos capaz de identificar las debilidades que pueden tener cada uno de estos activos. En cuanto a la evaluación de riesgos, se determina cual es el impacto que puede llegar a generar dicha vulnerabilidad y como esta afecta a la triada de seguridad de la información dentro de la organización.

Fase 2: Implementación de controles

La norma ISO 27001[5] establece hasta 113 puntos de control los cuales están divididos en:

- Políticas de seguridad de la información
- Controles Operacionales

Es justo recalcar que la norma es bastante flexible, es decir es posible modificar o agregar nuevos puntos de control según sea la necesidad de la organización. Sin embargo, estos puntos de control siempre deben estar dentro de lo que requiere la norma.

[5] https://www.isotools.org/pdfs-pro/iso-27001-sistema-gestion-seguridad-informacion.pdf

Fase 3: Definición de un plan de tratamiento de riesgos

Una vez que ya tenemos claro el análisis y evaluación de riesgos, en esta etapa se define un plan de tratamiento el cual contempla los conflictos que pueden causar estos riesgos. En otras palabras, esto quiere decir, establecer un plan que permita evaluar independientemente cada amenaza e identificar que tan crítica puede llegar a ser la misma.

Fase 4: Alcance de la gestión

Se determina el alcance para la implantación del SGSI según la dimensión de la organización, es decir, este alcance debe ser avaluado a partir de la cantidad de información que maneja la organización, cantidad de clientes, número de empleados, sucursales, etc.

Fase 5: Contexto de organización

En esta fase se realiza un análisis FODA[6] el cual permite determinar cuáles son las principales fortalezas y oportunidades, así como también reconocimiento de las debilidades y amenazas que posee la organización.

Fase 6: Partes interesadas

Las partes interesadas comprende a los clientes, proveedores de servicio de información, las Tics[7] y la sociedad en general.

Fase 7: Fijación y medición de objetivos

Dentro de esta fase se debe tener en cuenta que cada objetivo propuesto debe ser medible y estar sujeto a un indicador, el cual facilite un control del cumplimiento de cada actividad desarrollada. Además, estos objetivos deben ser comunicados a los empleados para que todos puedan trabajar en conjunto para conseguir un bien común.

Fase 8: Documentación

Como todo proceso de gestión es muy esencial la documentación de cada una de las actividades, por lo cual la norma ISO 27001 exige que la organización lleve un proceso de documentación tanto interno como externo para la administración de la información. Esta documentación puede estar descrita en papel, en digital o en archivos de audio o video.

Fase 9: Auditorías internas

Esta fase comprende la aplicación de auditorías que permitan llevar un control y mantenimiento del sistema, las mismas deben ser realizadas cada cierto tiempo. El objetivo de estas auditorías es determinar que el Sistema de Gestión de Sistemas de Información este cumpliendo con lo que dicta la legislación vigente y que cumpla con los propios objetivos del sistema.

[6] **A**menazas, **F**ortalezas, **D**ebilidades, **O**portunidades.
[7] Las tecnologías de la Información y Comunicación.

Luego de analizar la norma ISO 27001 y el Sistema de Gestión de Seguridad de la Información, nosotros proponemos la implementación de perímetros de seguridad en la red, mediante una arquitectura de red segura que permita mantener la disponibilidad, integridad y confidencialidad de la información, por lo que vimos necesario explicar estos temas.

## 3.2. Arquitectura Segura

Una arquitectura segura en una organización representa una barrera que separa fronteras, es decir, separa las zonas de red con el objetivo de proteger su información privada de atacantes internos o externos, en donde se pueda controlar cada detalle que afecte a su seguridad.

En la Ilustración 3, se representa los componentes que debe tener como mínimo una arquitectura segura.

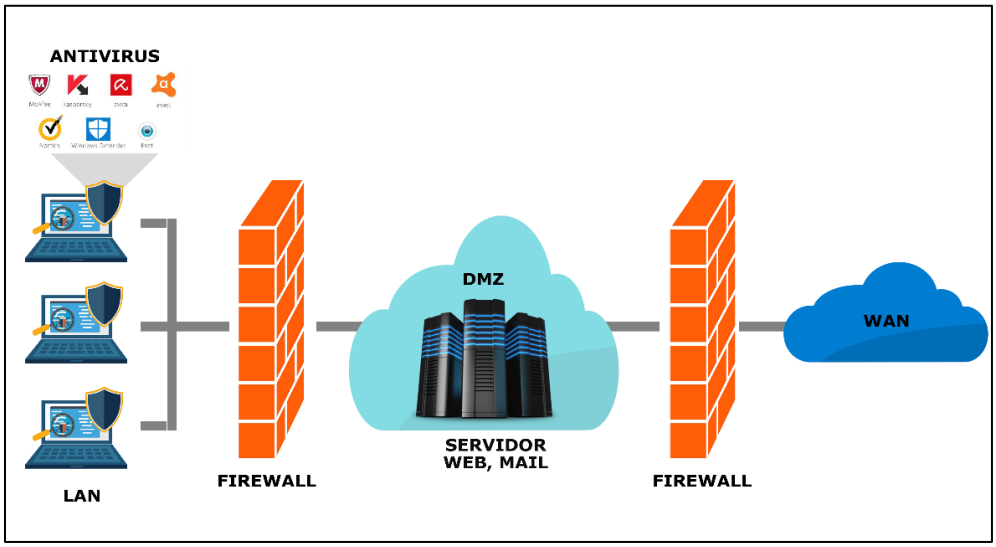


*Ilustración 3 Arquitectura básica de red segura.*

### 3.2.1. Firewall

Es un dispositivo de seguridad que se utiliza para supervisar y controlar el tráfico de entrada y salida en una red. También funciona como intermediario entre la red local y la red externa. Además, permite aplicar reglas de filtrado otorgando el acceso a un cierto tipo de tráfico, de igual manera se puede negar este tráfico [11]. Todo este proceso lo realiza con el fin de brindar protección a los equipos o servidores que estén conectados a la red brindando mayor seguridad y privacidad de la información de los usuarios [12].

Un Firewall permite controlar el acceso de entrada y salida de la red mediante mecanismos como:

**NAT (Network Address Translation):** Este tipo de traducciones sirve para comunicar redes totalmente distintas enmascarando una dirección IP privada para que esta tenga salida hacia el Internet con una dirección IP publica única que servirá solo para la comunicación interna hacia el exterior [13].

Este tipo de traducción de red puede ser:

– Estático: Una vez que existe una tabla de direcciones en donde se encuentra designado una dirección IP publica para cada IP privada, es decir, un host de una red privada podrá ser visible desde el Internet.
– Dinámico: En este caso existe un pool de direcciones que pueden tener cada IP privada en la red y estas direcciones pueden reutilizarse cuando estén disponibles.

**DNAT (Destination Network Address Translation):** esta traducción se utiliza cuando queremos exponer un servicio desde nuestra red privada hacia el exterior (Internet), en donde el enrutador NAT cambia la dirección IP de destino cambiando el encabezado de los paquetes IP de tal manera que redirecciona los paquetes que llegan desde el exterior (Internet) a una dirección IP interna virtual, para posteriormente llegar a nuestra dirección IP real [13], [14].

Un Firewall puede ser de Hardware y Software:

- Hardware

Suele ser un router de borde mediante los cuales tenemos salida hacia Internet, se encuentra situado entre la LAN y WAN, la mayoría de estos dispositivos son configurados por profesionales de las TI[8] debido a su complejidad de configuración. Estos permiten gestionar los paquetes entrantes y salientes para toda una red de computadoras [15], [16], [17].

- Software

Este tipo de Firewall pueden estar instalados en un ordenador, independientemente del sistema operativo, ayudan al control de paquetes entrantes y salientes solo de ese ordenador, es decir, si tenemos una red de computadores es necesario que en cada computador se encuentre instalado y cuando se encuentre alguna actualización se debe ejecutar el cambio en cada una de ellas [18].

Limitaciones de un Firewall

- Ningún Firewall puede controlar al 100% los ataques en una red.
- Un Firewall no impide que alguien interno en la red altere la información.
- El Firewall no es responsable de fallos en la red ni de los servicios que operan en la misma.
- El Firewall no otorga protección en la transferencia de algún programa o archivo infectado con algún tipo de Malware.
- El Firewall no otorga protección a ataques usando mecanismos de ingeniería social.

### 3.2.2. DMZ

Es una zona desmilitarizada ubicada entre la red LAN y la red WAN en donde se encuentran todos los servicios críticos de una empresa. El objetivo de esta zona es aislar los servicios públicos de los servicios locales, esta es utilizada en conjunto

[8] Tecnologías de la Información.

con el Firewall el cual mediante las reglas de filtrado permite bloquear o acceder la comunicación entre zonas [19].

Los servicios críticos dentro de una organización son:

- Servicio de DNS[9] (Resuelve direcciones IP[10] en nombres de dominio y viceversa)
- Servicio de Correo
- Servicio WEB
- Servicio de FTP (Transferencia de archivos bajo una arquitectura cliente-servidor)
- Base de datos, etc.

#### 3.2.3. Antivirus

Son programas informáticos cuyo objetivo es brindar protección, detectar posibles amenazas y eliminar los archivos o ejecutables que contengan código potencialmente maligno a causa de infección de algún tipo de virus informático, debido a que esto constituye en una amenaza para la seguridad y estabilidad de un sistema informático [20].

Tipos:

- **Preventivos:** Este tipo de antivirus deben estar instalados en el Disco Local C y están activos todo el tiempo, debido a que constantemente están monitoreando todo el tráfico en búsqueda de posibles virus. La principal desventaja de los antivirus preventivos es que debido al constante monitoreo consumen recursos de la computadora provocando que esta se haga un poco lenta.
- **Identificativos:** Identifican amenazas activas en la computadora producto del ataque de algún Malware que provoque un incorrecto funcionamiento o mal rendimiento del sistema.
- **Descontaminantes:** Como su nombre lo indica este tipo de antivirus se encarga de eliminar un Malware una vez este ya infecto el sistema. Su objetivo principal es eliminar la infección, sin embargo, de no ser esto posible intenta regresar hasta un punto de restauración antes de la infección.

## 3.3. Perímetros de Seguridad

La seguridad perimetral es un tema sumamente importante dentro de la seguridad de la red, debido a que en este se brindan o niegan el acceso a los recursos con los que cuenta una organización, para lo cual es importante tener en claro su concepto.

#### 3.3.1. Concepto de seguridad perimetral

En cuanto a seguridad perimetral, el autor Guijarro la define como uno de los métodos posibles de defensa de una red, en donde se basa en el establecimiento de recursos de seguridad en el perímetro externo de la red y a sus diferentes niveles. Esto permite definir niveles de confianza, brindando el acceso de determinados usuarios internos o externos a ciertos servicios y denegando el acceso a otros [21]. Por otro lado, Maiwald expone que son fronteras que separan la red de una

[9] Sistema de Nombres de Dominio.
[10] Conjunto de números que identifica, de manera lógica y jerárquica, a una Interfaz en red.

organización del mundo exterior. También se puede usar para separar partes o zonas de la red de una organización entre sí. Los perímetros pueden estar formados por componentes de red, dispositivos de seguridad (Firewalls) o mecanismos físicos como paredes y puertas [22].

La implementación de controles de seguridad perimetral incluye mecanismos tales como: Firewalls, software de detección de intrusos y VPN[11]. También puede incluir cambios en las arquitecturas de red. En conclusión, los diferentes perímetros de seguridad ayudan a evitar que los ataques de código malicioso que alteran el funcionamiento de una red, lleguen a nuestros centros de datos, minimizando riesgos en la seguridad [18].

## 3.4. Malware

Un Malware puede traer consecuencias como robo de información, lentitud del sistema, fallas en la red, etc. Por lo general son códigos maliciosos que buscan apropiarse de documentos importantes de una víctima. Este está estructurado por diferentes fases de funcionamiento. Se pueden clasificar según su estructura y la manera de propagarse, pueden darse en situaciones que palpamos diariamente en el flujo de la red.

### 3.4.1. Definición de Malware

Con respecto al termino Malware, los autores Escriva, Romero, Ramada y Pérez denominan como "software malicioso que puede modificar el funcionamiento de un equipo informático o alterar la información que procesa [4]". La corrupción del Malware puede manifestarse de diferentes maneras como; formatear el disco duro, eliminar o corromper archivos, robar información de inicio de sesión guardada, recopilar información confidencial (sus archivos y fotos privadas), enviando información sin nuestro conocimiento a terceras personas. Muchas variantes de Malware son sigilosas y funcionan silenciosamente sin el conocimiento del usuario. El término software malicioso tiene un ámbito más amplio que el de virus informático y se utiliza para designar a cualquier software que pueda representar una amenaza al sistema o resultar molesto para el usuario [23].

### 3.4.2. Fases de funcionamiento de un Malware

En la actualidad existe una gran cantidad de software malicioso, cada uno con diferentes objetivos. Sin embargo, el funcionamiento de cada uno de ellos sigue un mismo patrón que hemos decidido separar en fases que se exponen en la Ilustración 4.

[11] Red Privada Virtual.

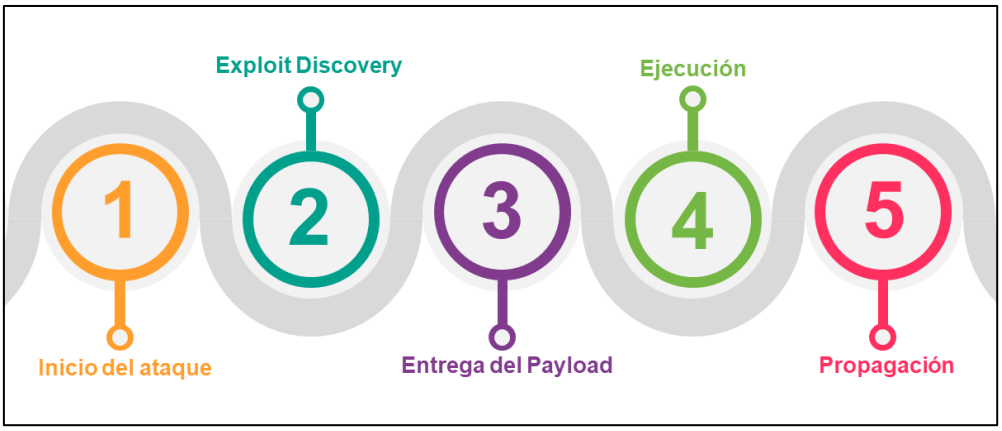


*Ilustración 4 Fases funcionamiento del Malware.*

Fase 1: Ataque dirigido e inicio del ataque

Determinar un método especifico de ataque según sea su objetivo final, es decir, primero se debe tener claro que vamos a atacar para de esta manera poder escoger el tipo de ataque que pueda cumplir con este objetivo. Por ejemplo, podemos utilizar Phishing mediante correo electrónico, incitando a los destinatarios a abrir un archivo que puede ir adjunto al correo.

Fase 2: Exploit Discovery

En esta etapa el Malware es empaquetado en Exploit Kit. Los Exploit Kit son sentencias de código que permiten aprovechar algún tipo de vulnerabilidad que aún no ha sido detectada por el sistema operativo.

Muchos atacantes insertan los Exploit Kit en sitios web que se muestran para el usuario como paginas legitimas. Cuando la víctima accede a la página web, el Exploit empieza a analizar el sistema de la víctima y extrae; información del sistema operativo, versiones de programas, navegadores, identificando algún tipo de vulnerabilidad que exista y que pueda servir de provecho para el atacante.

Fase 3: Entrega del Payload

Dentro de esta etapa el atacante crea el Payload y mediante algún método de ingeniería social hace que el cliente descargue el ejecutable sin que se dé cuenta que es software maligno, la efectividad de este dependerá de las diferentes técnicas de ofuscación que utilice el atacante para ocultar su Malware.

Fase 4: Ejecución del ataque

En la fase anterior únicamente se descargó el Malware, sin embargo, aún no está en marcha, en esta fase la victima inicia el archivo infectado provocando que se ejecute el Malware en segundo plano realizando la tarea para la que fue diseñado (robar información personal, permitir el acceso remoto, etc.) infectando el sistema operativo. La gran mayoría de antivirus puede detectar la ejecución del Malware, sin embargo, si el atacante utilizo herramientas de ofuscación será poco probable que este sea detectado.

Fase 5: Propagación del Malware

Finalmente, al no ser detectado y eliminado por el antivirus este empieza a

propagarse en el sistema operativo, al llegar al objetivo final este establece una comunicación con el atacante enviando la información recopilada de la víctima o a su vez crea una puerta trasera para que el atacante puede administrar remotamente los recursos de la víctima.

### 3.4.3. Clasificación

#### 3.4.3.1. Virus informático

Los virus informáticos son software malicioso que corrompen los archivos del sistema, causando alteraciones en el funcionamiento del equipo infectado, pueden alterar carpetas, ocultar archivos y en los casos más grabes pueden llegar a modificar los registros del sistema para evitar la detección de Firewall. Es necesario recalcar que los virus no se ejecutan por sí solos por lo que necesitan que la víctima lo ejecute. Los virus informáticos se instalan en la memoria RAM y desde ahí infectan los archivos ejecutables; toma uno de estos archivos, lo infecta y lo vuelve a guardar en el disco duro, provocando que cada vez que el usuario abra el programa, se ejecute el virus. Si por algún motivo el usuario se da cuenta cual es el programa infectado, lo desinstala y no reinicia la computadora, el virus seguirá alojado en la memoria RAM [4], [24].

#### 3.4.3.2. Gusano

En cuanto a un gusano, es un Malware que se aprovecha de las conexiones de red para difundirse como recursos compartidos de una red local, canales de chat, correo electrónico, redes P2P[12], etc. Estos se propagan automáticamente duplicándose a sí mismos. Su finalidad es consumir los recursos del sistema y colapsar la red de comunicaciones de un sitio provocando su caída. Se debe agregar que en las redes P2P un equipo puede descargarse archivos de cualquier otro equipo de la red y a su vez compartir archivos con los demás; por ello, todos los equipos pueden ser a la vez clientes y servidores. Esta es una de las vías preferidas de infección, puesto que los gusanos se camuflan como archivos con nombres atractivos para los usuarios, adoptando nombres de películas de actualidad o vídeos humorísticos. Por otra parte, en un correo electrónico se pueden presentar como mensajes adjuntos camuflados dentro del código HTML, por lo que basta con previsualizar el mensaje para activarlos. En todos estos casos, los mensajes que los incluyen suelen tener un asunto interesante para captar la atención del destinatario y hacer que abra el mensaje [4], [25].

#### 3.4.3.3. Spyware y Adware

Según Bettany y Halsey [26], exponen que un Spyware es un software espía que trabajan con o sin conexión a Internet, estos se encargan de recopilar información del equipo infectado y enviarla hacia el atacante, la información que este puede recolectar es muy amplia, todo depende de la cantidad de información que cuente el equipo infectado, para lo más común que se utiliza los spyware es la obtención de las pulsaciones de teclado que el usuario pulsa cuando inicia sesión en sitios web, tiendas online y bancos, utilizando un Keylogger[13] [4].

En cuanto a los adwares, los autores Bettany, Halsey, Nihad y Rami exponen

[12] Red Peer-to-peer

[13] Software o hardware que captura las pulsaciones del teclado del equipo infectado.

que este tipo de Malware son utilizados para mostrar anuncios en su PC, comúnmente vienen en forma de ventanas emergentes que se despliegan cuando se navega a través de Internet. Al igual que los Spyware estos son utilizados para recopilar información del usuario y de la PC, pueden venir adjuntos en software legitimo o complementos que se agregan a los navegadores. Estos Malwares no representan una amenaza real como tal, a menos que lleven un Payload adicional, como un Keylogger [4], [26], [27].

#### 3.4.3.4. Ransomware

Un Ransomware es un Malware de mayor complejidad y peligrosidad que cifra y bloquea el acceso al Sistema Operativo y amenaza con borrar o publicar los datos personales del usuario si no se realiza un pago. Una vez que un usuario paga el rescate, el hacker envía la clave de descifrado al usuario. Sin embargo, no existe una garantía de que envíe la clave de descifrado. Lo más común es que los atacantes piden que el pago sea en criptomonedas para no ser rastreados [27]. Al igual que los demás tipos de Malware este puede adjuntarse a un software legitimo por lo que no es detectado.

#### 3.4.3.5. Backdoor

Con respecto a backdoor o puerta trasera podemos indicar que es un tipo de Malware, el cual permite a un atacante obtener acceso remoto a todos los recursos de su máquina. Se denomina puerta trasera a cualquier punto débil de un programa o sistema mediante el cual una persona no autorizada puede acceder. Las puertas traseras pueden ser errores o fallos, o pueden haber sido creadas a propósito, por los propios autores, pero al ser descubiertas por terceros, pueden ser utilizadas con fines ilícitos. También, "se consideran puertas traseras a programas que, una vez instalados en el ordenador de la víctima, dan el control de éste de forma remota al ordenador del atacante [28]". También, Sikorski y Honing [29], exponen que el código de puerta trasera a menudo implementa un conjunto completo de capacidades, por lo que, cuando se usa este tipo de ataque no es necesario descargar código o Malware adicional. Las puertas traseras se comunican a través de Internet de muchas maneras, pero un método común es a través del puerto 80 utilizando el protocolo HTTP [4].

#### 3.4.3.6. Troyano

El siguiente punto explica la función de los troyanos, según Nihad y Rami [27], este tipo de Malware puede infectar la computadora silenciosamente. Por lo general, se instala como parte de una instalación de software legítimo, trabajando sigilosamente en segundo plano y no son detectables por los antivirus. Los troyanos pueden obtener acceso a todas las funciones de su sistema, incluida la cámara y el micrófono, tienen la capacidad de eliminar archivos y monitorear sus actividades en línea y las pulsaciones de teclas.

Un troyano, normalmente está constituido por dos programas: un cliente en el equipo atacante, que es el que envía las órdenes, y un servidor que se instala en el ordenador infectado y es el que recibe las órdenes del intruso y las ejecuta, enviando la información solicitada [4].

Por otro lado, Costas [7], indica que un troyano es un código malicioso con capacidad de crear una puerta trasera o backdoor que permita la administración remota a un usuario no autorizado. Pueden llegar al sistema de diferentes

formas, las más comunes son: descargado por otro programa malicioso, al visitar una página web insegura, dentro de otro programa que simula ser inofensivo, etc. Los troyanos no corrompen archivos o programas y, a diferencia de los gusanos, no tienen la capacidad de propagarse automáticamente [4], [30].

#### 3.4.3.6.1. R.A.T (Remote Administration Trojan)

En cuanto a un R.A.T. se define como una herramienta de gestión que se utiliza para administrar de forma remota una o un conjunto de computadoras. Los R.A.T. a menudo se utilizan para realizar ataques dirigidos con objetivos específicos, permitiendo acceder remotamente a los equipos infectados dando la experiencia como si tuviera acceso físico al equipo de la víctima [31]. Dicho lo anterior, el funcionamiento de un R.A.T. consta de dos partes; el servidor y la víctima. En la computadora de la víctima se ejecuta un Payload (un Payload posee una dirección IP, un puerto y una contraseña, las mismas que se deberán colocar en el servidor para establecer la comunicación) que debe ser generado con anterioridad por el servidor. Finalmente, el servidor tiene toda la unidad de control con todas las funcionalidades del R.A.T. como, por ejemplo; administración de archivos, escritorio remoto, acceso al shell del sistema, acceso a la webcam, etc.

Un R.A.T. puede infiltrarse en su computador mediante métodos de ingeniería social como; correo electrónico, anuncios falsos que sin querer usted pudo haber autorizado, permitiendo que se inyecte en su sistema. Una vez inyectado en su computador, un R.A.T. es capaz de evadir los análisis de antivirus, editando los registros del sistema como System.ini permitiendo activarse de manera silenciosa en cada reinicio.

## 3.5. Métodos de propagación de un Malware

### 3.5.1. Ingeniería Social

Existen diferentes opiniones acerca de la definición de ingeniería social y cuál es su función. Según Hadnagy [32], la ingeniería social es el arte o la ciencia de manipular las acciones futuras que puede tomar los seres humanos en algún aspecto de su vida cotidiana. “Es el arte de manipular a las personas, mediante engaño, para que den información o realicen una acción [33]”. Por otra parte, las técnicas de ingeniería social son utilizadas para obtener acceso a información de naturaleza sensible para el usuario, en donde un atacante utiliza la influencia y la persuasión que involucra una acción relacionada con la computadora, para engañar a las víctimas y aprovechar su confianza para obtener información privilegiada [28], [34].

Tomando en cuenta lo dicho en el párrafo anterior, concluimos que la ingeniería social cubre la aplicación de un conjunto de técnicas de engaño o persuasión, con el fin de obtener información sensible de una persona particular o un empleado de alguna compañía, con el fin de que sobrepase los límites para cumplir un objetivo, o que a su vez realice algún tipo de acción que pueda llegar a generar una falla de seguridad que posteriormente será aprovechado por los atacantes. Además, se puede decir que la única forma realmente efectiva de mitigar la amenaza de ingeniería social es mediante el uso de políticas y tecnologías de seguridad combinadas, que establezcan reglas básicas para el comportamiento de los empleados, educación y capacitación apropiada. No es posible crear una metodología para contrarrestar la ingeniería social, debido a que cada persona actúa de forma diferente ante un

ataque. Sin embargo, se pueden identificar rasgos que podemos moderar y predecir su comportamiento ante un ataque de ingeniería social.

Según Mann [33], es posible implementar una metodología de seguridad para protegerse contra las debilidades humanas dentro de la ingeniería social.

1. Identificar los riesgos de seguridad relacionados con vulnerabilidades humanas, a través de un análisis de sus sistemas.
2. Detectar vulnerabilidades humanas a través de pruebas sistemáticas.
3. Compartir información para comprender las debilidades humanas que los atacantes pueden explotar.
4. Desarrollar contramedidas: personas capacitadas con capacidad de detectar y contrarrestar un ataque; y, mejoras sistémicas efectivas para reducir la dependencia de las personas y sus debilidades.

Por lo general, un atacante utiliza el conocimiento de las debilidades humanas para establecer un ataque que puede llegar a ser funcional, debido a que un ataque de Ingeniería Social no tiene garantía de que funcione, por el contrario, conlleva un alto riesgo para el atacante. Sin embargo, debido a las diferentes tecnologías de comunicación y herramientas de anonimato es más fácil garantizar su protección.

#### 3.5.1.1. Ataques de Ingeniería Social

Los principales ataques de Ingeniería social son:

- Phishing

Es el tipo de ataque más común para realizar Ingeniería social, este se basa en la suplantación de identidad, en donde el atacante tiene como objetivo conseguir información confidencial del usuario tales como; credenciales de inicio de sesión, datos de tarjetas de crédito, etc. Este tipo de ataque se da cuando un atacante, por medio de correo electrónico se hace pasar por una entidad de confianza, en donde indica al usuario descargarse algún archivo adjunto o que redirija a través de un enlace a un sitio web determinado, en el cual este realizó una suplantación de nombre de dominio del sitio web original, creando así un sitio web falso con la misma apariencia del original, lo que permite que el usuario no se percate de que está introduciendo su información personal en un sitio fraudulento lo que podría conllevar consecuencias devastadoras como robo de su dinero o identidad [35], [36].

- Pretexting

El Pretexting trata de generar la mayor empatía y confianza con las víctimas para conseguir su información personal y posteriormente usarlo sin su consentimiento, para obtener la información privada los atacantes suelen hacerse pasar por empleados de la empresa requiriendo una confirmación de la información. Posteriormente a la obtención de la información el atacante realiza una llamada al servicio de atención al cliente haciéndose pasar por el titular que contrato el servicio burlando la seguridad que tiene la gestión telefónica en una empresa [37], [38].

- Baiting

En este tipo de ataque, el atacante entrega una recompensa para atraer a la víctima, por ejemplo: permite descargar música, videos, películas, etc. gratis a

cambio de que el usuario se autentifique a través de alguna red social, para validar sus credenciales. Este ataque no solo comprende a páginas web, también se puede realizar de manera física con un propósito diferente, un ejemplo muy sencillo y muy utilizado por los atacantes es el uso de memorias USB infectadas con algún tipo de Malware. Los atacantes generan empatía con algún empleado con acceso a información sensible y le regalan la memoria USB como un obsequio [3], [38].

- Grooming

Este tipo de ataque se basa netamente en la ingeniería social, en donde el atacante suele ser un adulto que busca aprovecharse de la confianza de un menor de edad para que este le entregue a cambio información sobre el circulo en el que se rodea hasta contenido de tipo sexual. El atacante se hace pasar por una persona que no es, tomando la identidad de otra persona, esté estudia a la víctima, le genera confianza y forma un vínculo de amistad en el cual se cree con la potestad de pedir a la víctima cualquier petición hasta cumplir con su objetivo y obtener lo que buscaba [37], [38].

### 3.5.2. Ofuscación

#### 3.5.2.1. ¿Qué es la ofuscación?

Por lo que se refiere a ofuscación de software, los autores Beron, Henriques, Varanda y Uzal [39], mencionan que la ofuscación de software es una técnica para ocultar el flujo de control del software, así como las estructuras de datos que contienen información sensible. También se utiliza para mitigar la amenaza de la ingeniería inversa, también conocida como reingeniería. En otras palabras, es una técnica que se utiliza para empaquetar y comprimir un programa malicioso permitiendo que este no sea detectado. Teniendo en cuenta que un Malware casi siempre codifica transmisiones de formas únicas, la codificación y la ofuscación no solo los ayudan a evitar las firmas de detección, sino que también ocultan el verdadero objetivo del Malware. Esta técnica puede ser tan simple como convertir cadenas a hexadecimales o tan sofisticada como desarrollar algoritmos personalizados para traducciones detalladas [29], [40].

#### 3.5.2.2. ¿Qué es un Crypter?

A su vez un Crypter según Barria, Cordero, Cubillos y Osses es un software que permite proteger la información realizando una encriptación de los datos para ocultar el contenido real de un conjunto de datos para que los mismos no sean encontrados de manera directa por un usuario [41]. En el momento que los datos están cifrados es posible filtrar a través de los diferentes perímetros de seguridad sin ser detectados, para que posteriormente sea descifrado y cumpla con su función habitual. Un Crypter puede ser utilizado tanto para ocultar información o como también para que evitar que un Malware sea detectado por un software antivirus [42].

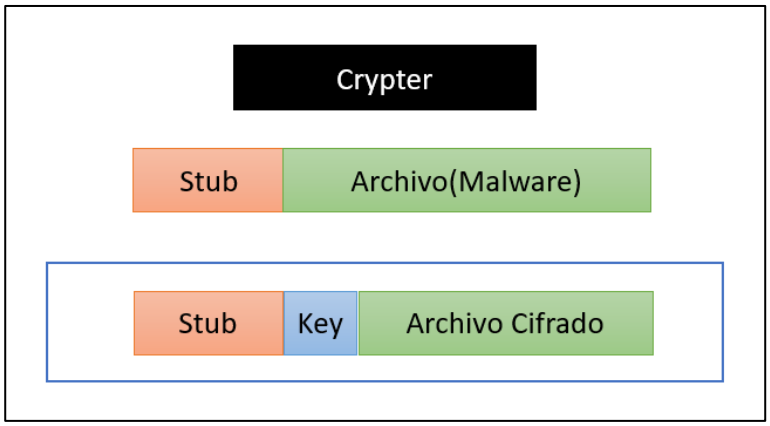


*Ilustración 5 Estructura interna de un Crypter.*

En la Ilustración 5 se muestra la estructura interna de un Crypter y a continuación se detalla la función de cada elemento.

**Crypter:** Se encarga de la ejecución del algoritmo de cifrado al archivo de entrada.
**Stub:** Es la parte más esencial que poseen los Crypter's, este se encarga de descifrar y ejecutar el archivo que fue cifrado con anterioridad.
**Archivo:** Puede ser un programa o un ejecutable de cualquier tipo que queremos cifrar.
**Key:** Esta llave puede ser generada o introducida por el usuario, varía según cada Crypter.

Tipos

- **Scantime:** El archivo no es detectado por los antivirus hasta que este sea ejecutado. Cuando se ejecuta es detectado automáticamente por los antivirus debido a que se descompone el código fuente.
- **Runtime:** Este tipo de Crypter posee un cifrado mucho más complejo, producto de esto es mucho menos probable de ser detectado por la mayoría de antivirus en el momento de ejecutarlos.

Algunas herramientas de Crypter que podemos encontrar son: Captus Joiner, Dr True Crypt, Root Crypter, TNT Crypter, Metasploit Crypter entre otras que están alojadas en el Sistema Operativo Bugtraq. Estas herramientas permiten ofuscar código malicioso ocultándolo a través de una imagen en donde nos permite establecer en que ruta se guardará y si al interactuar con este archivo se debe ejecutar de manera inmediata, siendo indetectable para algunos antivirus. Estas herramientas funcionan de manera eficaz en versiones de Windows XP y 7. Una recomendación eficaz y gratuita es Cobra Crypter el mismo que esta actualizado para trabajar con las versiones más actuales de Windows, permitiendo pasar inadvertido ante ciertos antivirus como Windows defender, McAfee, etc.

### 3.5.3. Esteganografía

La esteganografía es el arte de ocultar datos en archivos de audio, imágenes, videos u otro archivo de texto, de tal manera que un mensaje pueda ser enviado y pase desapercibido. Si es interceptado por un tercero, al tomar la apariencia de un archivo normal, este no sabrá el mensaje oculto, solo las personas involucradas en la comunicación sabrán como leer el mensaje oculto. Algunas técnicas de

esteganografía se utilizan para ejecutar diferentes tipos de virus, el usuario puede descargarse de sitios web imágenes, videos, audios infectados y al momento que dan clic estos pueden estar ejecutando un virus sin percatarse de que en segundo plano puede ocurrir [43], [44].

## 3.6. Métodos de protección del Malware

### 3.6.1. Métodos activos

Los métodos activos comprenden los diferentes mecanismos de seguridad tales como; los antivirus y los Firewalls dedicados a proteger una organización ante alguna amenaza interna o externa mediante la aplicación de protocolos de seguridad y filtrado de ficheros.

### 3.6.2. Métodos pasivos

Por otro lado, los métodos pasivos son aquellos principios de sentido común de cada persona, por ejemplo; evitar utilizar software de dudosa procedencia, no pasar por alto alertas de seguridad del antivirus, no descargar ni instalar software pirata, analizar siempre los dispositivos extraíbles, no abrir correos de personas anónimas, no utilizar un medio de comunicación seguro con agentes externos, etc.

## 3.7. Criptografía

La criptografía es un algoritmo que se utiliza para establecer un canal de comunicación seguro, mediante la verificación y validación de llaves secretas entre cliente y servidor sin la necesidad de intermediarios.

### 3.7.1. ¿Qué es Criptografía?

En cuanto a la Criptografía esta se denomina como la ciencia de la comunicación secreta. Se utiliza para diseñar e implementar técnicas de comunicación confiable entre dos partes. “Es un algoritmo criptográfico que funciona en combinación con una clave para cifrar y descifrar datos. Esta clave está compuesta por una cadena de bits [27].” Cuanto más extensa y compleja sea la clave de cifrado que se utilice, es poco probable que un atacante pueda romperla [45].

### 3.7.2. Tipos

Según Nihad y Rami [27], exponen dos tipos de criptografía:

#### 3.7.2.1. Criptografía simétrica

También conocido como criptografía de clave secreta (SKC), en este tipo de cifrado, tanto el remitente como el receptor utilizan la misma clave para cifrar y descifrar los datos. La principal desventaja de este esquema es que toda la operación depende de una sola clave. Si la clave se ve comprometida por una parte no autorizada, el sistema puede ser vulnerado.

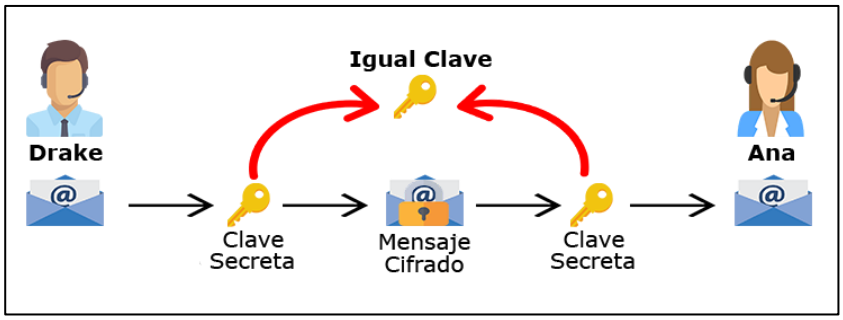


*Ilustración 6 Criptografía simétrica.*

3.7.2.2. Criptografía asimétrica

Criptografía de clave pública (PKC), este tipo utiliza diferentes claves para el cifrado (clave publica) y descifrado (clave privada). Las dos claves están matemáticamente vinculadas, sin embargo, no es posible derivar la clave de descifrado de la clave de cifrado.

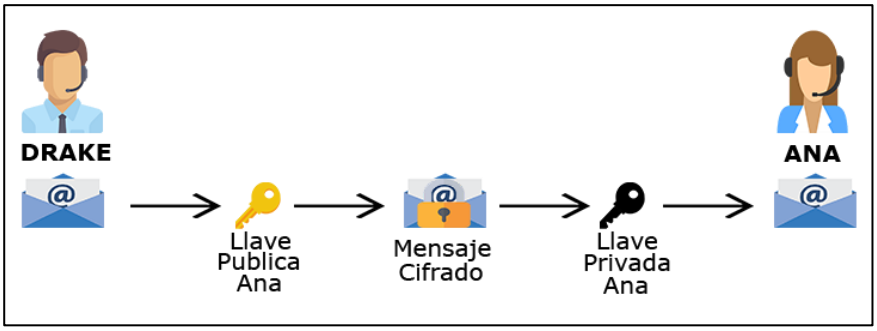


*Ilustración 7 Criptografía asimétrica.*

3.7.3. Algoritmos Criptográficos

HASH: Un hash es un algoritmo matemático, es decir es una secuencia alfanumérica que se genera a través de la codificación de un texto, archivo o documento. A su vez este puede ser fijo, único e irrepetible en donde no importa el tamaño de los datos de entrada, el valor del hash siempre será el mismo.

AES (Advanced Encryption Standard): Es un algoritmo matemático funciona como un bloque de cifrado iterativo y simétrico, utiliza como clave de cifrado que puede ser 128,192 o 256 bits de largo para cifrar una comunicación de datos.

MD5 (Message-Digest Algorithm 5): Es una función de Hash que se utiliza para verificar la integridad de los datos por lo general este está compuesto por un archivo de texto.

## 3.8. Sistema Operativo Objetivo

En la actualidad existen una extensa diversidad de sistemas operativos los mismos que son utilizados para diversas funcionalidades, sin embargo, el sistema operativo dominante en el mercado, en las empresas y en los hogares es sin duda Windows.

Windows sufre una gran cantidad de ataques tales como; Ransomware, Troyanos, Spyware, Keylogger, robo de credenciales, integración en redes de Bots y DDOS. Esta información hemos tomado de Instituto Nacional de Ciberseguridad[14] el cual nos brinda información de los ataques y vulnerabilidades de seguridad que se han encontrado en Windows, además brinda avisos de nuevas amenazas.

Uno de los ataques más grandes que ha sufrido fue el ataque de Ransomware denominado WannaCry[15] el mismo llego a afectar a más de 200.000 computadoras en 150 países**,** restringiendo el acceso a sus equipos a cambio de un rescate, WannaCry se aprovechó de la vulnerabilidad MS17-01 también conocida como EternalBlue[16], la cual permitía aprovecharse de que ciertas versiones del protocolo SMB (protocolo de compartición de archivos, impresoras, etc.) que podían ser engañadas fácilmente, para recibir paquetes de datos de lugares remotos fuera de la red. Otro de los ataques más actuales que fue descubierto a tiempo por Microsoft es BlueKepp[17] catalogado con un gusano informático el cual pretendía aprovechar la vulnerabilidad CVE-2019-0708 (servicios de escritorio remoto) para esparcirse.

Existen diferentes tipos de ataques de troyanos que se han desarrollado en Windows con el fin de espiar u obtener la información privada de la víctima, por lo general engañan a las víctimas a través de ingeniería social para que ejecuten el troyano en sus sistemas. Debido a lo puntualizado anteriormente y teniendo en cuenta que los ataques de troyanos como los R.A.T. han tenido éxito al infectar Windows hemos decidido trabajar con este sistema operativo, aprovechándonos de vulnerabilidades que ya han sido descubiertas.

[14] https://www.incibe.es/
[15] https://www.bbc.com/mundo/noticias-39929920
[16] https://www.cisecurity.org/wp-content/uploads/2019/01/Security-Primer-EternalBlue.pdf
[17] https://www.us-cert.gov/ncas/alerts/AA19-168A

# 4. Trabajos Relacionados

Durante el proceso de investigación realizamos una revisión bibliográfica para el desarrollo de nuestro proyecto, en donde se revisó algunos paper's y libros que nos han servido para tener una estructura conceptual sobre nuestro trabajo y citamos los de mayor interés.

Por una parte, Iker, Murat, Marchetti, Pierazzi y Colajani explican que un R.A.T, permite ver, modificar los archivos y funciones del usuario en el sistema, monitorear y registrar la actividad del usuario, y usar el sistema de la víctima para atacar otros sistemas. Además, afirman que un R.A.T. pueden ocultarse fácilmente en el sistema con sus métodos avanzados de infección y pueden estar presentes como entidades fantasmas sin ser detectados por algún software de seguridad. Además, definen que los R.A.T. no son solo ataques simples, las agencias de inteligencia y los grupos activistas también los usan para propósitos específicos, como chantajear y espiar. Ellos proponen como métodos de propagación el uso de las redes sociales, mensajes de correo electrónico u agregar el R.A.T. a programas legítimos que el usuario instalara en su sistema [46] [47].

Por otra parte, los autores Samuel, Graham y Hinds [48], en su artículo denominado "Hunting Malware: An Example Using Ghost" demuestran cómo se puede detectar un RAT mediante el análisis de las evidencias o residuos que este deja después del ataque. Ellos proponen un escenario en el cual pueden comprobar que esto sea posible, el escenario consta de tres máquinas; un equipo con el sistema operativo Windows 7 el cual va a actuar como víctima, un equipo atacante y un equipo con Kali Linux el cual se encarga del análisis forense. Mediante Wireshark detectan el tráfico de red que genera Ghost RAT, capturan los paquetes que se generan en el momento exacto que la víctima activa la comunicación con el equipo atacante, logrando detectar el puerto con el que están trabajando y sus paquetes, una vez realizado este proceso en el equipo con Kali se realizan diferentes búsquedas de la existencia de la firma de Ghost RAT logrando tener éxito. Finalmente, en el equipo de la víctima buscan la existencia de un archivo winlog.exe el cual solo debe estar alojado en el directorio C:/Windows/Systems32, afirman que si existe un duplicado del archivo winlog.exe este equipo está infectado con el Malware indicando una posible amenaza. Esta investigación tomamos como referencia para darnos cuenta que para la detección de un Malware en el caso específico un RAT es necesario tener una cierta base de conocimiento en cuanto al manejo de herramientas de monitoreo del tráfico de red, manejo de herramientas de Kali Linux, manejo de comandos de Linux, análisis forense lo cual un usuario común no los posee lográndonos aprovechar de esta ventaja.

Asimismo, los autores Yamada, Morinaga, Unno y Takenaka [49], exponen la detección de actividades maliciosas basadas en RAT en redes empresariales, en la cual en el análisis del comportamiento de las diferentes familias de RAT identifican dos tipos de RAT, un tipo de conexión directa (acceso desde el Internet) y un tipo de conexión inversa (cliente-servidor). Los de tipo conexión directa todo el ataque es realizado desde el servidor del atacante mientras que los de conexión inversa es necesario que la víctima realice algún tipo de acción para activar el ataque. Además, durante su proceso de investigación y pruebas de los 43 RAT de conexión inversa identifican un grupo de características comunes las cuales son: conexión entre el formulario del servidor y el cliente, comunicación encriptada, comunicación push y verificación del estado de conexión cada cierto tiempo. Tomando en cuenta estos factores y diferentes parámetros que muestran en el desarrollo de su investigación decidimos crear un prototipo de software RAT de conexión inversa debido a que las empresas por seguridad tienen bloqueado los accesos externos a la red mientras que con un RAT de conexión inversa tenemos mucha más posibilidad de tener existo con la propagación y activación por parte de la víctima.

De igual manera, los autores Maarof y Shaid [50], en su artículo denominado “Malware behavior image for Malware variant identification” explican que el comportamiento del Malware hace referencia a lo que hace, exhibe y causa la ejecución de este al sistema operativo afectado. Consideran que para representar el comportamiento del Malware es conveniente tomar en cuenta los siguientes aspectos: monitorear los cambios que presenta el sistema operativo en el instante que es ejecutado, capturar la secuencia de llamadas API, captura de solicitudes de paquetes IRP y monitoreo de la actividad de la red del Malware. Es por ello que proponen una técnica para visualizar el comportamiento del Malware a través de imágenes. Esta técnica consta de 4 etapas: captura del comportamiento del Malware, mapeo del comportamiento a color, mapa de colores y finalmente la generación de la imagen.

Siguiendo este enfoque, el autor Tangen [51], en su artículo denominado “Detecting Remote Administration Trojans through Dynamic Analysis using Finite-State Machines” expone que los RAT pueden ser bastante específicos y solo pueden extenderse a ciertas personas o subgrupos, sin tener la amplia cobertura de un gusano replicante o una campaña de spam. También define que se pueden crear RAT completamente nuevos para su propio uso personal, lo que hace que sea muy poco probable que algún investigador de Malware lo tenga en sus manos. Además, explican que se utilizan tecnologías de ofuscación como mecanismo para prevenir el análisis y la detección de los RAT de los antivirus comerciales existentes. El análisis de Malware mediante la observación estática de su código fuente puede ser fácilmente engañado por técnicas de ofuscación como el cifrado y la inserción de operaciones adicionales.

Por último, los autores Shigemoto, Fujii, Kuriama, Kito, Nakakoji, Fujii y Kikuchi [52], en su artículo denominado “Development of White List Based Autonomous Evolution of Defense System for RAT Malware” proponen un sistema que permite evaluar la conexión HTTP de un RAT basándose en un enfoque de lista negra y lista blanca, dentro de la lista negra gestionan todos los servidores a los cuales se intentó comunicar y se bloqueó las comunicaciones del Malware, mientras que en la lista blanca están únicamente los servidores seguros. Ellos proponen la implementación de un mecanismo de autenticación adicional orientado a la lista blanca debido a que consideran que se usa para los negocios, el sistema permite contrarrestar el Malware sin interrumpir las actividades comerciales. Cuando el servidor de conexión no está dentro de los servidores seguros (lista blanca) este debe pasar por un segundo mecanismo de autenticación (CAPTCHA), si el usuario pasa la autenticación el sistema permite la solicitud HTTP caso contrario la bloquea evitando que se realice la comunicación entre el cliente y el servidor del RAT impidiendo el robo de información.

# 5. Fases de desarrollo del Proyecto

En este ítem procedemos a indicar el proceso que seguimos para el desarrollo de nuestra tesis y de nuestra herramienta de acceso remoto R.A.T, además la propuesta de diseño de software y también se documenta los escenarios de prueba para el respectivo análisis de resultados.

## 5.1. Análisis de Requerimientos

El proceso de análisis y obtención de requerimiento se llevó a cabo a base del estudio desarrollado a las empresas y personas que trabajan dentro del área de las telecomunicaciones, los mismos tienen la constante necesidad de un software que permita administrar de forma remota los recursos de un equipo que está alojado en otra localidad, con el fin de optimizar tiempo y dinero, así como también dar soporte

temprano a fallas concurrentes otorgando un porcentaje alto de disponibilidad en el sistema administrado. Nuestro sistema permitirá cubrir directamente esta necesidad, brindando un software RAT que trabaja bajo una arquitectura Cliente – Servidor que cubre la gestión de archivos, escritorio remoto, acceso al shell del sistema, etc.

- Requisitos Funcionales

El software debe permitir administrar remotamente las siguientes funciones:

1. Escritorio remoto.
2. Shell remoto.
3. Administración de archivos.
4. Acceso al administrador de tareas.
5. Acceso a información del equipo de la víctima.

- Requisitos no Funcionales

1. El sistema debe brindar un software amigable con el usuario.
2. La aplicación no debe tener permisos de administrador, únicamente debe tener acceso a los permisos de un usuario común.
3. La aplicación debe ser compatible con las últimas versiones de Windows.
4. El sistema debe ser capaz de mostrar alertas cuando se conectan nuevos clientes.
5. El sistema debe contar con un manual de usuario para su correcta utilización.

## 5.2. Diseño de software

- Diagrama de casos de uso

✓ Generación del cliente

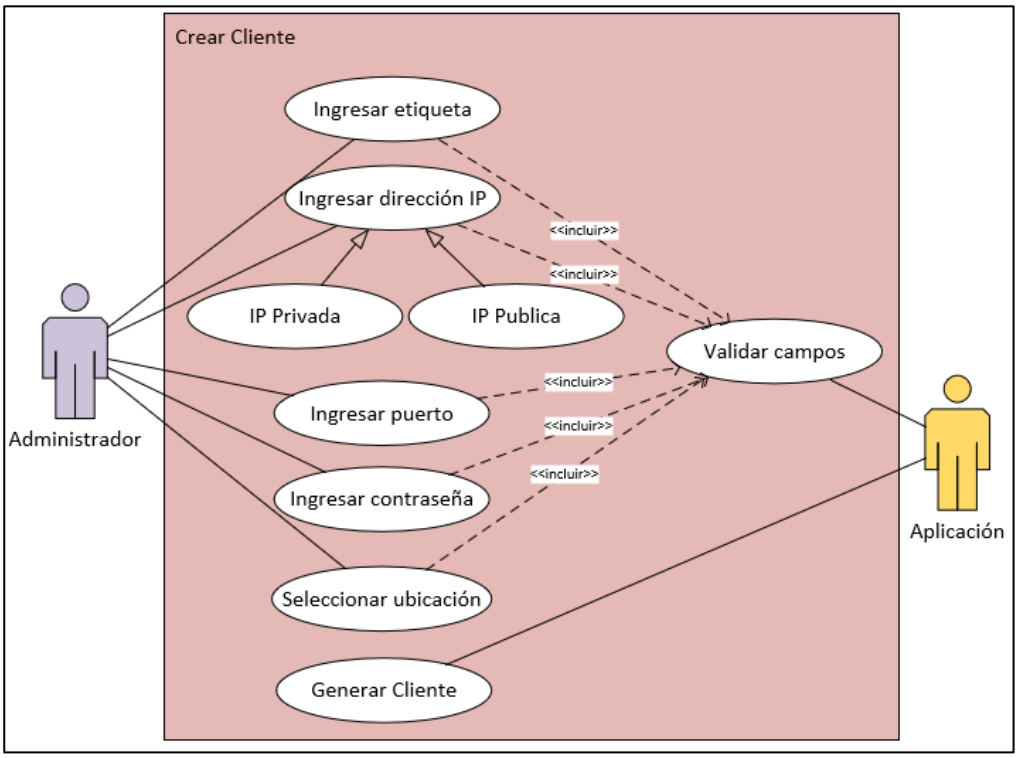


*Ilustración 8 Evento crear cliente.*

✓ Captura de nuevas conexiones

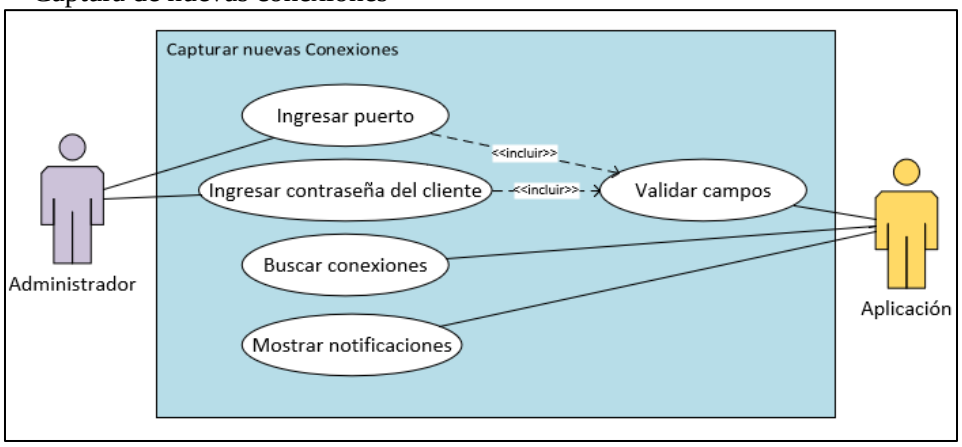


*Ilustración 9 Evento Capturar conexiones.*

✓ Gestión de herramientas de administración remota

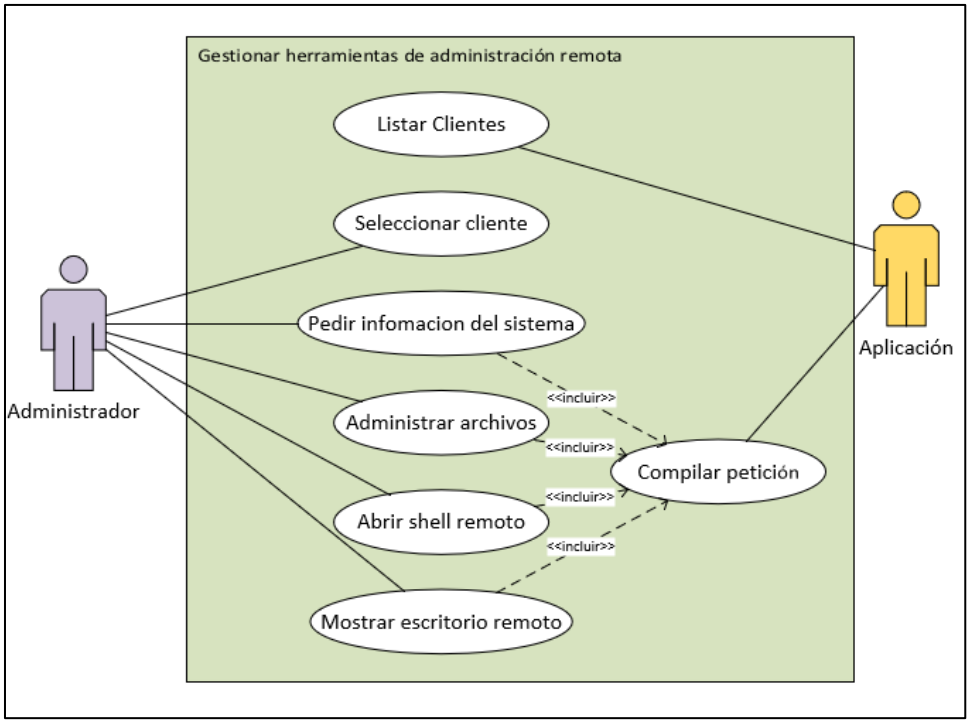


*Ilustración 10 Evento Gestionar herramientas de administración remota.*

- Diagrama de Estados

El diagrama expone los cambios de estado de la aplicación adjuntando en el Anexo 8.1.

- Entorno de Desarrollo

El software está desarrollado en Visual Studio en el lenguaje de programación C# bajo el entorno .NET Framework, este Framework cuenta con un gran motor de formularios, bibliotecas y una documentación completa en donde se encuentra como puede ser utilizado cada comando, al ser propio de Windows este nos facilita el

acceso y manejo de todos los recursos que necesitamos controlar en nuestro software R.A.T.

## 5.3. Implementación

Dentro de la implementación del proyecto hemos planteado probar nuestro software en tres diferentes escenarios y analizar su comportamiento en cada uno de ellos, independientemente de los mecanismos de seguridad que se utilicen. Para el adecuado uso del software hemos realizado un manual de usuario el cual podemos encontrar en el Anexo 8.2.

### 5.3.1. Escenario 1

En el primer escenario decidimos probar nuestro software en una arquitectura de red sencilla, en la cual la víctima no va a contar con ningún mecanismo de seguridad (Firewall, Antivirus) que le pueda proteger ante algún tipo de ataque como se puede observar en la Ilustración 11.

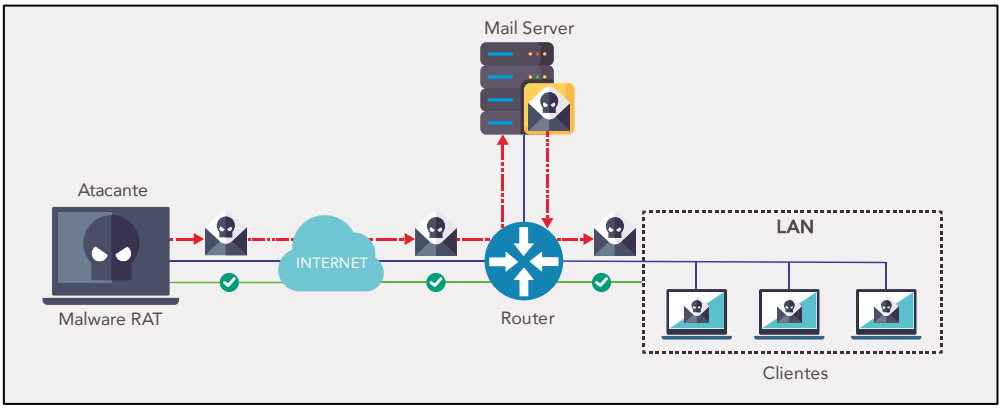


*Ilustración 11 Escenario 1.*

Componentes del escenario:

- Windows 10 (Servidor R.A.T.).
- Windows 7 (Victima).
- Enrutadores Mikrotik.
- Ubuntu (Servidor de Correo).

Dentro de este escenario utilizamos dos enrutadores Mikrotik para emular una red WAN, también se virtualizó un servidor de correo que servirá como mecanismo de comunicación entre el atacante y la víctima en la cual mediante la técnica de ingeniería social persuadimos a la víctima enviándole un correo que lleva adjunto el ejecutable de nuestra aplicación.

A continuación, en la Ilustración 12, se muestra la dirección IP del Servidor y en la Ilustración 13 la dirección IP de la Victima con los cuales vamos a trabajar.

✓ Dirección IP del Servidor

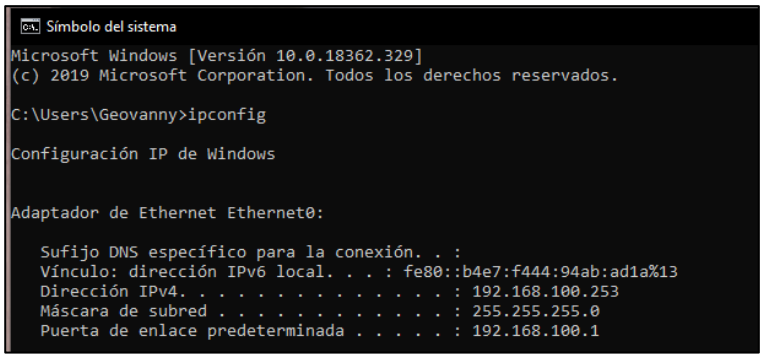


*Ilustración 12 Dirección IP del servidor.*

✓ Dirección IP de la Victima

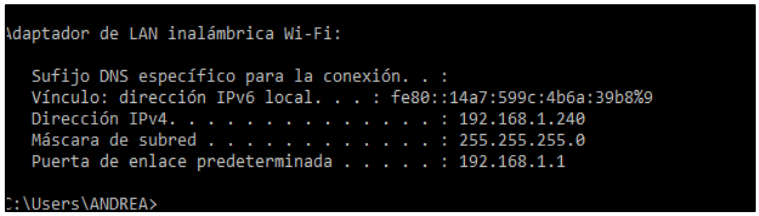


*Ilustración 13 Dirección IP de la víctima.*

Como se puede evidenciar tanto la dirección IP de la víctima, como la dirección IP del servidor son distintas debido a que las dos pertenecen a redes privadas. Tomando esto en cuenta, desde el equipo atacante abrimos la aplicación y procedemos con la creación del cliente, como se ve en la Ilustración 14.

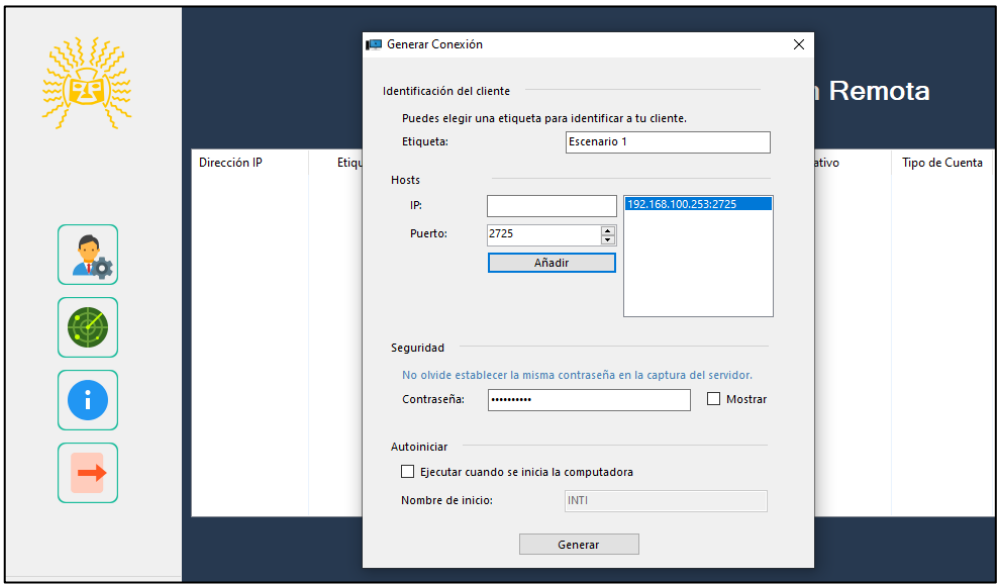


*Ilustración 14 Ventana crear cliente.*

Una vez llenado todos los campos y que estos hayan sido validados, generamos el archivo .exe el cual permite comunicar el cliente con el servidor. Posteriormente

abrimos la pestaña capturar e ingresamos el puerto y contraseña por donde queremos hacer una búsqueda de nuevos clientes, como se muestra en la Ilustración 15.

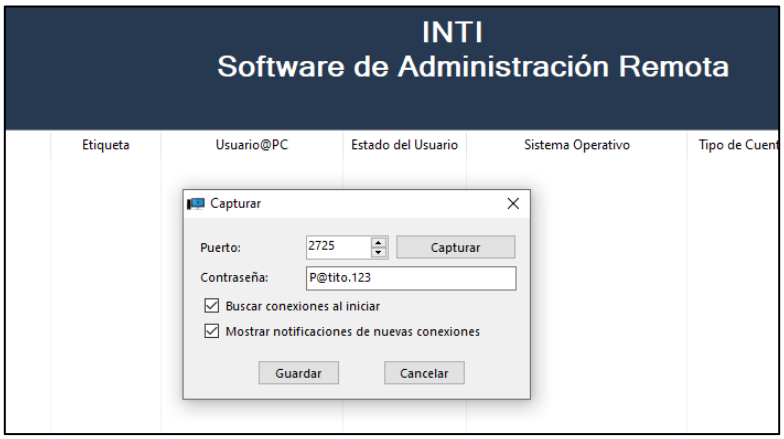


*Ilustración 15 Capturar clientes mediante puerto.*

Como se mencionó en el marco teórico, la ingeniería social tiene como objetivo engañar a una persona para aprovecharse de la misma, hemos decidido utilizar el Pishing como técnica de propagación. Como se puede observar en la Ilustración 16, nos hacemos pasar por un compañero de trabajo para que la víctima no desconfié que le adjuntamos un Malware y realice las acciones que se le piden en el correo

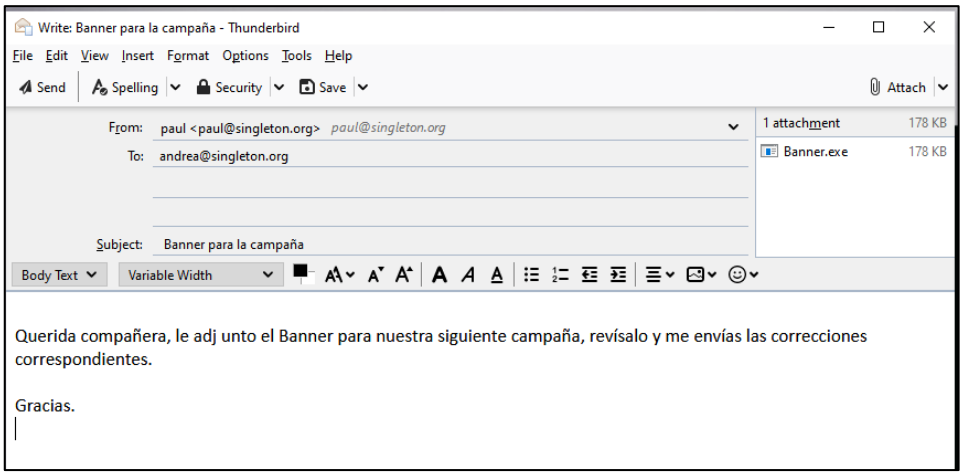


*Ilustración 16 Ingeniería social mediante email.*

Dada la posibilidad de que todo salió según lo planificado: la victima recibe el mensaje de correo electrónico, lo leerá pensando que es de confianza y para el bien de la organización descarga el archivo ejecutable que supuestamente es un banner, lo intenta abrir y provoca que el Malware se ejecute sin recibir alerta alguna de que su equipo ha sido infectado.

*Nota: Para los dos siguientes escenarios se utilizará la misma técnica de propagación del Malware y creación del Cliente, pasos empleados en este escenario por lo cual no se especificarán los mismos.*

### 5.3.2. Escenario 2

Como se ve en la Ilustración 17, en este segundo escenario probaremos si el software funciona cuando la víctima ya forma parte de una organización, en donde

se encuentra protegida por el Firewall propio del sistema operativo y por un router de borde (ENDIAN Firewall), el cual actúa como una barrera entre la LAN y la WAN, el mismo cuenta con reglas de filtrado de tráfico de entrada y salida, bloqueo de puertos, NAT, etc. Lo que buscamos probar en este escenario es demostrar que nuestra aplicación es capaz de aprovecharse de los puertos que están abiertos en una empresa, para realizar un ataque hacia los usuarios de la LAN a pesar de que ellos ya cuenten con protección del Firewall. Para pruebas de este escenario utilizamos el puerto 443 correspondiente al protocolo HTTPS.

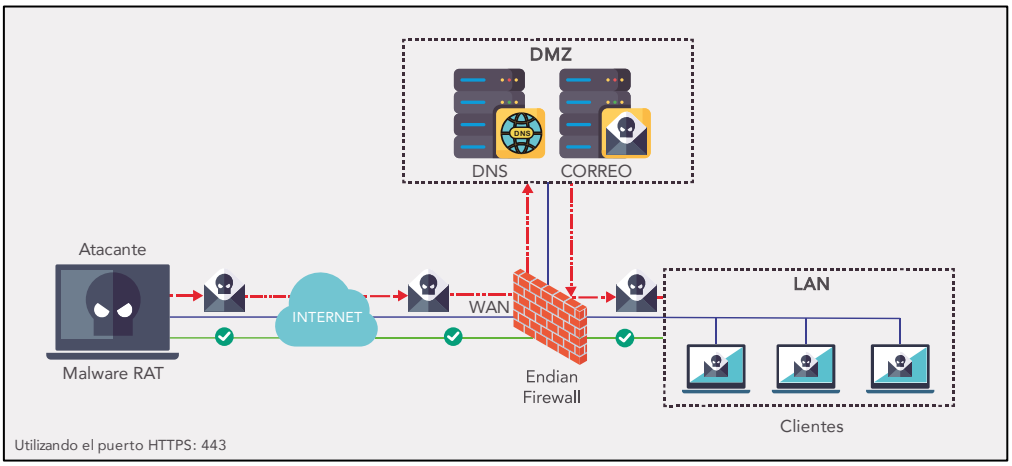


*Ilustración 17 Escenario 2.*

Componentes del escenario:

- Windows 10 (Servidor R.A.T.).
- Windows 7 con Firewall habilitado (Victima).
- Windows 10 con Firewall habilitado (Victima).
- ENDIAN (Configurado con reglas de filtrado de entrada y salida de tráfico).
- Ubuntu (Servidor de Correo).

Evidencias del escenario 2:

En la Ilustración 18, se puede observar una captura de pantalla de la dirección IP del Servidor.

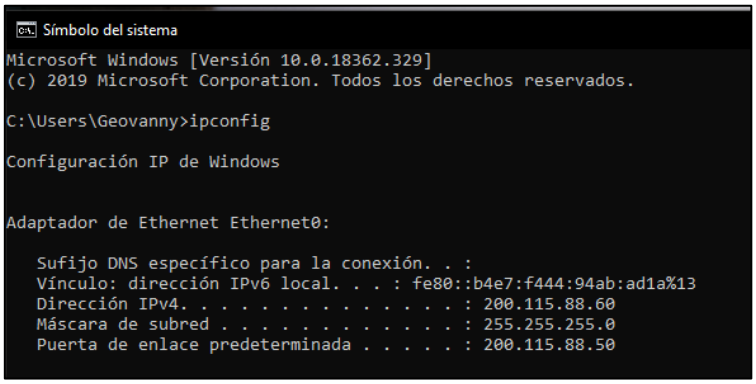


*Ilustración 18 Dirección IP del servidor.*

En este escenario realizaremos pruebas con víctimas que tengan el Sistema Operativo Windows 7 y Windows 10. Como se puede observar en la Ilustración 19,

tenemos una captura de pantalla de Windows 7 y en la Ilustración 20, tenemos una captura de Windows 10.

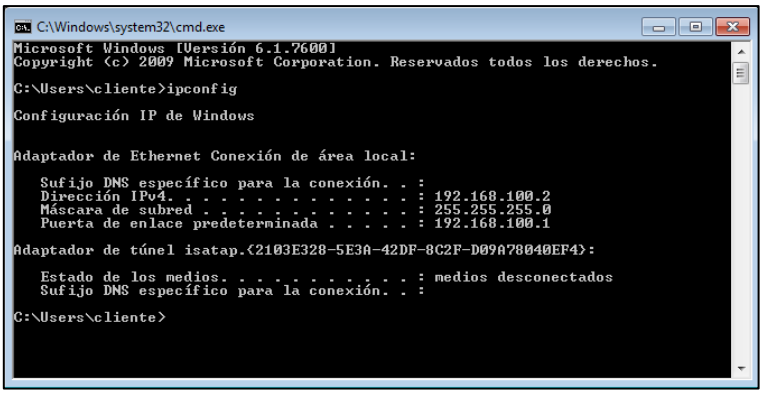


*Ilustración 19 Dirección IP cliente Windows 7.*

- ✓ Dirección IP de la víctima con Windows 10

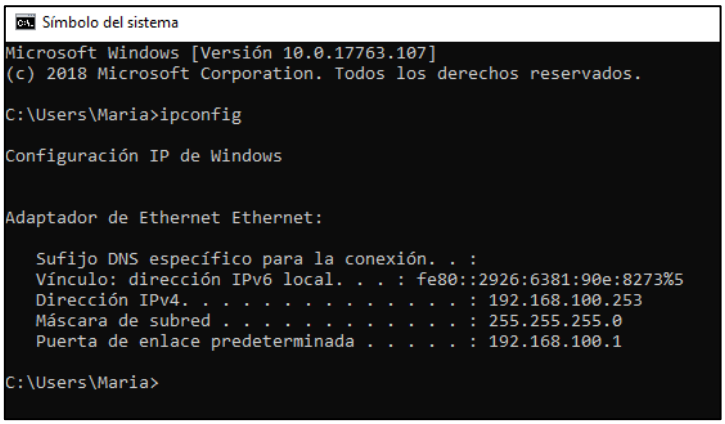


*Ilustración 20 Dirección IP cliente Windows 10.*

Como se puede evidenciar en la Ilustración 21, tenemos activado el Firewall de Windows 7 en el equipo de la víctima.

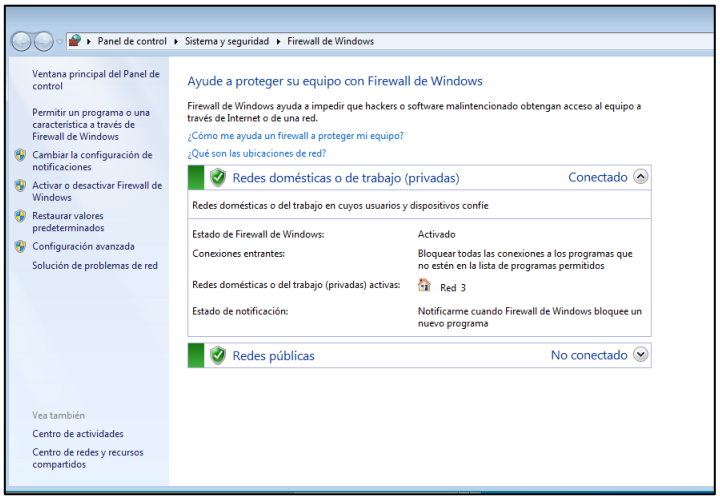


*Ilustración 21 Firewall Windows activado.*

De igual manera lo activamos en Windows 10 como se ve en la Ilustración 22.

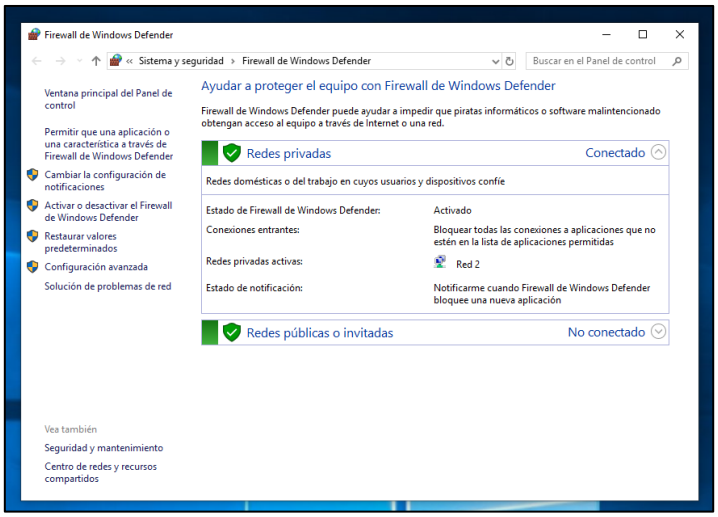


*Ilustración 22 Firewall Windows 10 activado.*

✓ Creación del Nuevo Cliente

Dentro de la creación del nuevo cliente, se debe tener muy en claro la dirección IP local del servidor del atacante y el número de puerto en el que vamos a trabajar. Esto tenemos en cuenta debido a que en un ambiente empresarial el Firewall solo abre los puertos que utilizan sus aplicaciones y todos los demás puertos se bloquean. Como se puede observar en la Ilustración 23, utilizamos el puerto 443 que por lo general está abierto al público.

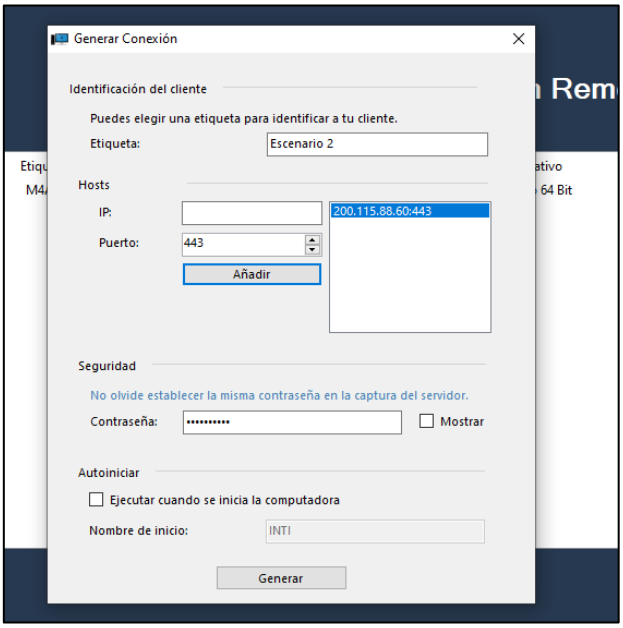


*Ilustración 23 Creación cliente utilizando el puerto 443.*

✓ Ofuscación de Software

El método de ofuscación lo utilizamos para propagar el Malware a equipos que utilizan el sistema operativo Windows 10, debido que su versión de Windows Defender ya cuenta con un motor de búsqueda de Malware mucho más sofisticado debido a sus últimas actualizaciones.

La herramienta que utilizamos para ofuscación se llama “Cobra Crypter”, esta herramienta nos permite que un cierto número de antivirus no detecten nuestro Malware, además como se puede ver en la Ilustración 24, podemos crear un ensamblado totalmente personalizable dándonos la libertad de generar un ejecutable que tenga la apariencia de un software de confianza para que la víctima no tenga ninguna sospecha de que es un software ilegítimo.

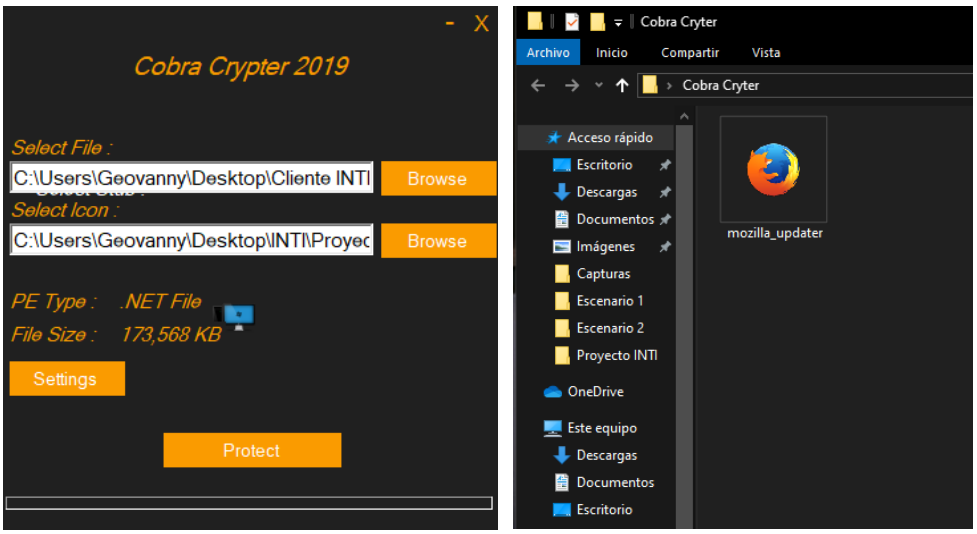


*Ilustración 24 Cobra Crypter.*

- ✓ Despliegue Endian Firewall

Endian protege tres zonas: LAN, DMZ y WAN. Dentro de la red LAN tenemos ubicado los clientes y en la DMZ los servicios críticos DNS y Correo. Como queremos probar si es posible realizar un ataque externo a los clientes que están protegidos, el equipo atacante estará ubicado en la red WAN, utilizando una IP pública. En la Ilustración 25, se muestra una captura de pantalla de la ventana de administración del Firewall Endian.

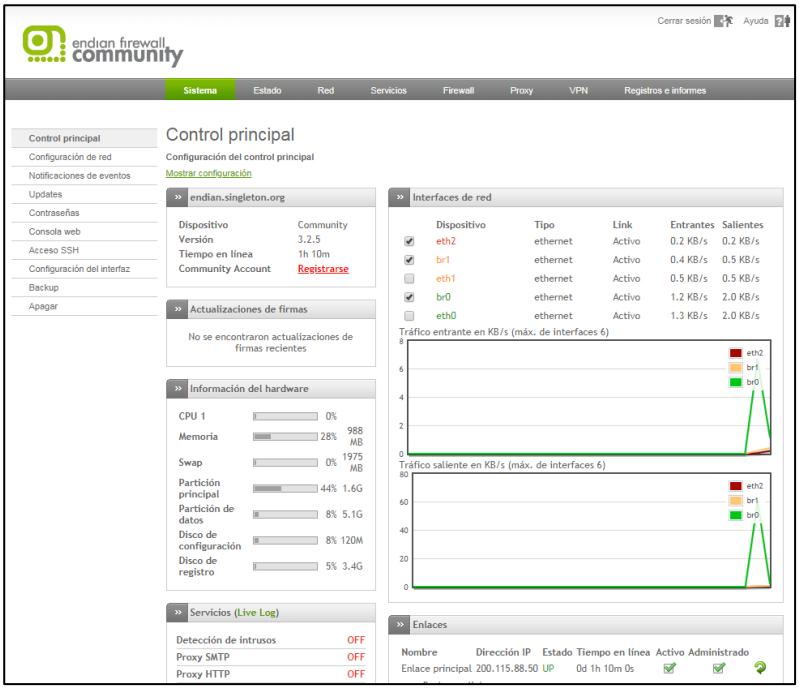


*Ilustración 25 Interfaz gráfica de Endian.*

Como se puede observar en la Ilustración 26 dentro del tráfico de salida únicamente tenemos permitido los protocolos HTTPS, IMAP, SMTP y DNS para permitir el uso del servicio de correo entre la LAN y WAN. Como se puede observar en la Ilustración 26, dentro del tráfico de salida únicamente tenemos permitido los protocolos HTTPS, IMAP, SMTP y DNS para permitir el uso del servicio de correo entre la LAN y WAN. En redirección de puertos habilitamos el acceso de la WAN solo al servicio de correo de la DMZ y negamos todo lo demás.


*Ilustración 26 Reglas de trafico de salida.*

De igual manera, como se puede observar en la Ilustración 27, para la propagación del Malware utilizamos Ingeniería Social enviando adjunto nuestro Malware.


*Ilustración 27 Aplicación de Ingeniería social.*

Por último, una vez que las victimas den clic sobre el Malware y activen la comunicación con el servidor, estos aparecerán en el panel de administración del R.A.T., como se puede observar en la Ilustración 28.


*Ilustración 28 Conexión con los clientes usando el puerto 443.*

### 5.3.3. Escenario 3

La topología de este escenario es similar al escenario 2, solo que esta vez la victima contará con un mecanismo de seguridad más. Como se ve en la Ilustración 29, los clientes se encuentran protegidos con McAfee Antivirus Plus, el Firewall propio del sistema operativo y el router de borde (ENDIAN Firewall). En este caso al contar la víctima con un antivirus es más probable que nuestro software R.A.T. sea detectado, debido a que cada Antivirus ya cuenta con una gran lista de registros de detección de troyanos similares a nuestro proyecto, convirtiéndose en un reto que debemos evadir para que se pueda llevar a cabo la ejecución del R.A.T.

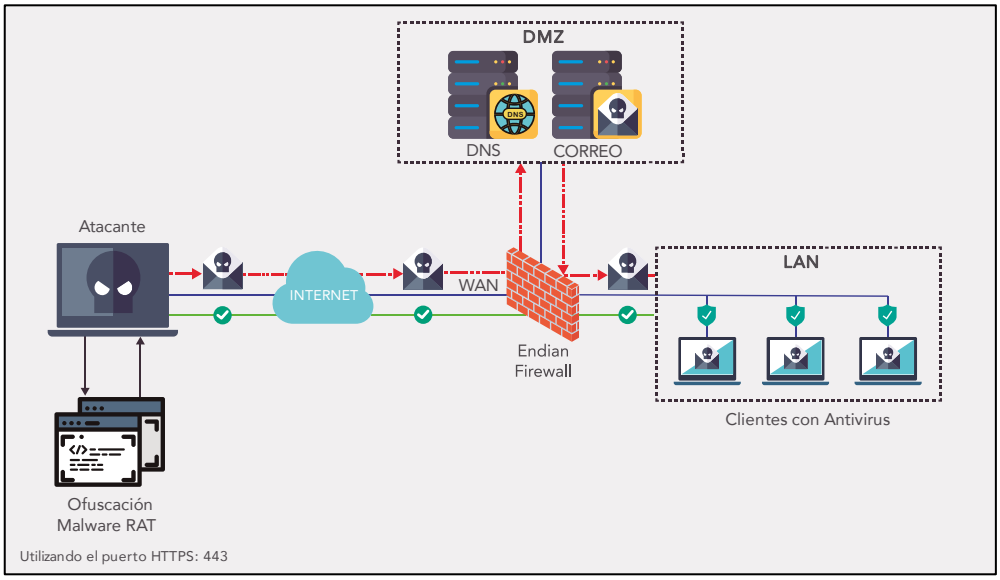


*Ilustración 29 Escenario 3.*

Componentes del escenario:

- Windows 10 (Servidor R.A.T.)
- Windows 10 con Firewall habilitado y Windows Defender (Victima)
- ENDIAN (Configurado con reglas de filtrado de entrada y salida de tráfico).
- Ubuntu (Servidor de Correo)

En la Ilustración 30, se muestra la dirección IP del Servidor.

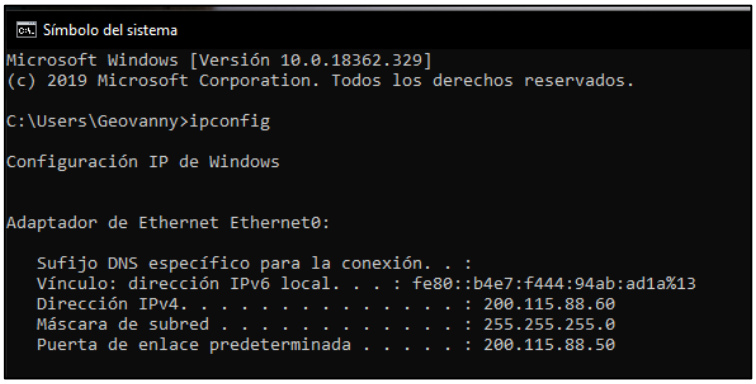


*Ilustración 30 Dirección IP del servidor.*

De igual manera, en la Ilustración 31, se muestra la dirección IP de la víctima la cual tiene una red totalmente distinta a la del servidor.

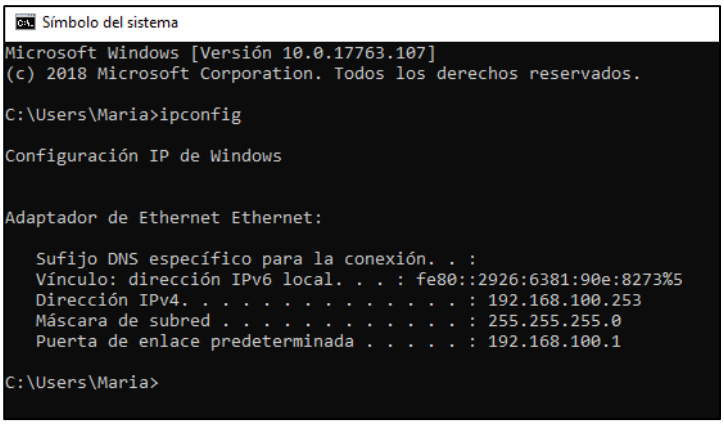


*Ilustración 31 Dirección IP de la víctima.*

Asimismo, como se ve en la Ilustración 32, para el proceso de ofuscación se utilizó Cobra Crypter ya que la víctima cuenta con mecanismos de seguridad que podrían detectar y eliminar el Malware.

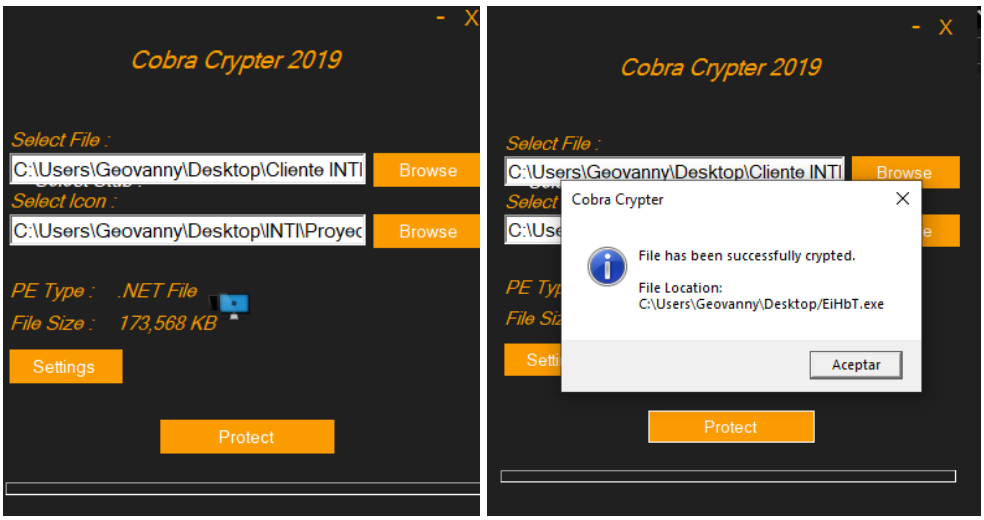


*Ilustración 32 Ofuscación del Malware con Cobra Crypter.*

Al igual que los dos escenarios anteriores el método de propagación, como se ve en la Ilustración 33, sigue siendo el mismo ya que es el método más utilizado y más efectivo para enviar un Malware.


*Ilustración 33 Ingeniería social.*

Como se puede observar en la Ilustración 34, la victima tiene instalado y ejecutando McAfee Antivirus Plus en la versión 16.0.

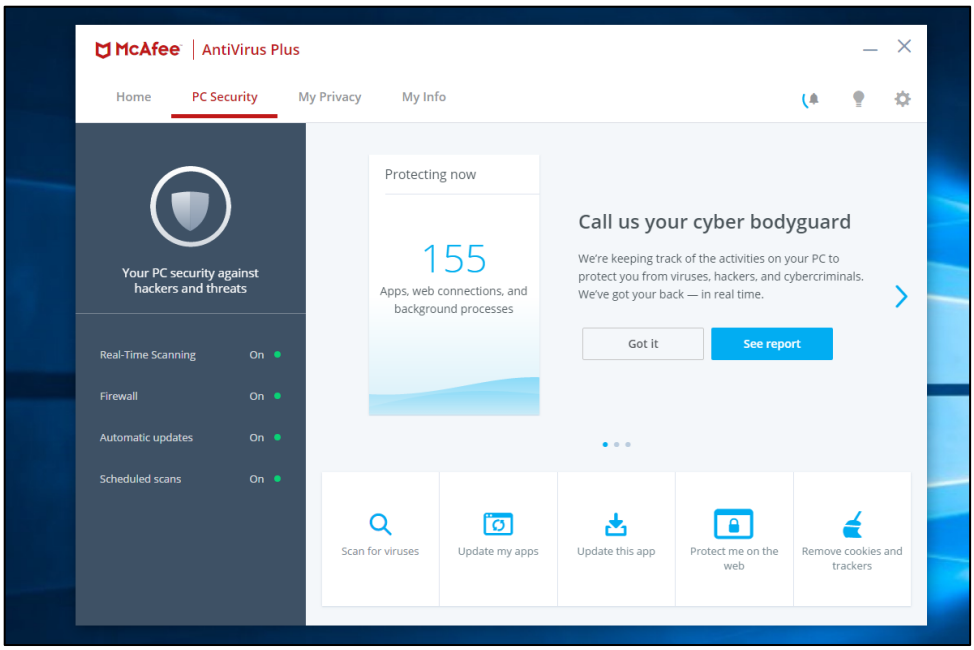


*Ilustración 34 Antivirus McAfee Plus.*

En el momento que la víctima recibe el archivo con el Malware y realiza un análisis en busca de virus, el antivirus no lo detecta, como se ve en la Ilustración 35.

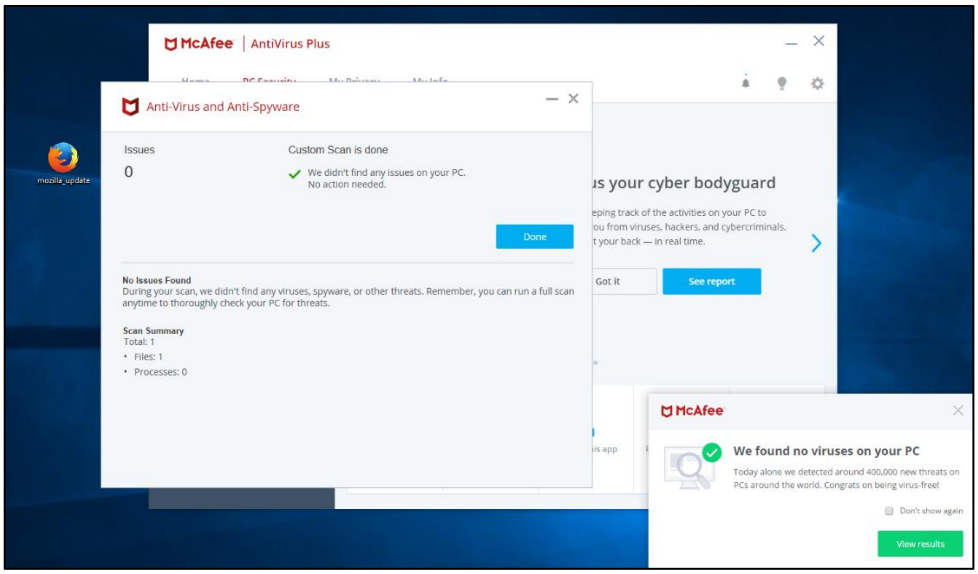


*Ilustración 35 Análisis del R.A.T.*

Al no ser detectado como Malware la victima ejecuta el archivo, provocando que este se ejecute en segundo plano. Sin embargo, como se ve en la Ilustración 36, el proceso se encuentra ejecutando dando la apariencia de un proceso de Firefox.

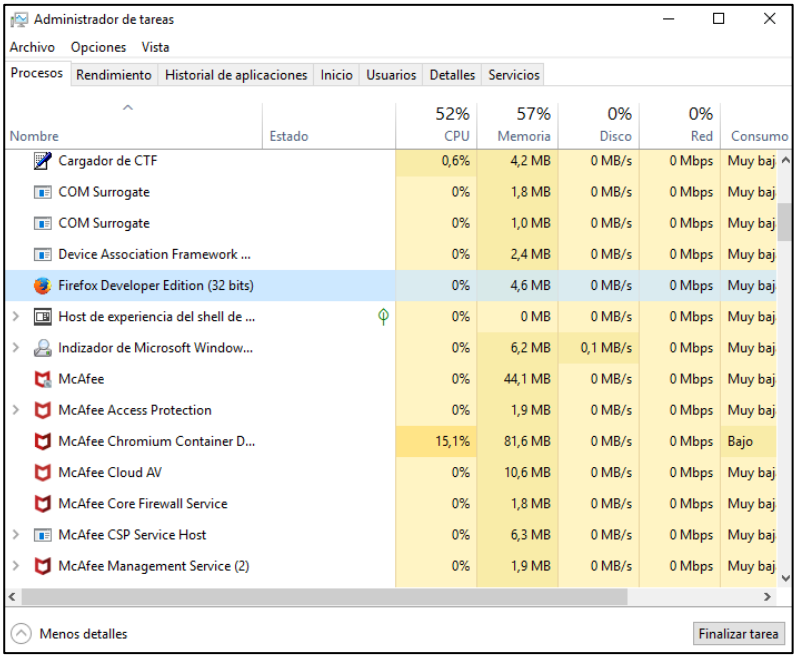


*Ilustración 36 Ejecución del Malware en el administrador de tareas.*

Como se indicó en la Ilustración 36, el proceso ya se encuentra ejecutando correctamente en el equipo de la víctima por lo que en el panel de administración ya se lista el equipo conectado, como se ve en la Ilustración 37.

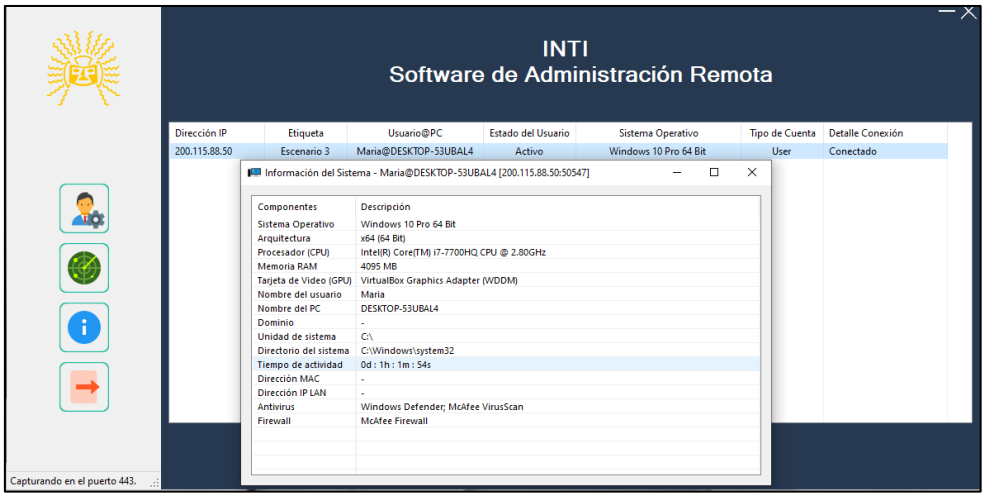


*Ilustración 37 Conexión cliente-servidor usando el puerto 443.*

## 5.4. Análisis de Resultados

### 5.4.1. Resultado del Escenario 1

Como resultado del primer escenario se logró tener éxito en la comunicación con la víctima, el R.A.T. fue ejecutado por la víctima sin percibir que lo que estaba ejecutando era un Malware, esto era predecible debido a que su equipo no contaba con ningún mecanismo de seguridad que le advirtiera la presencia de algún software que podría llegar a alterar el funcionamiento del sistema.

A continuación, en la Ilustración 38 Información sistema del cliente. se muestra la información del equipo de la víctima, datos como; la versión del sistema operativo, memoria RAM, procesador, etc.

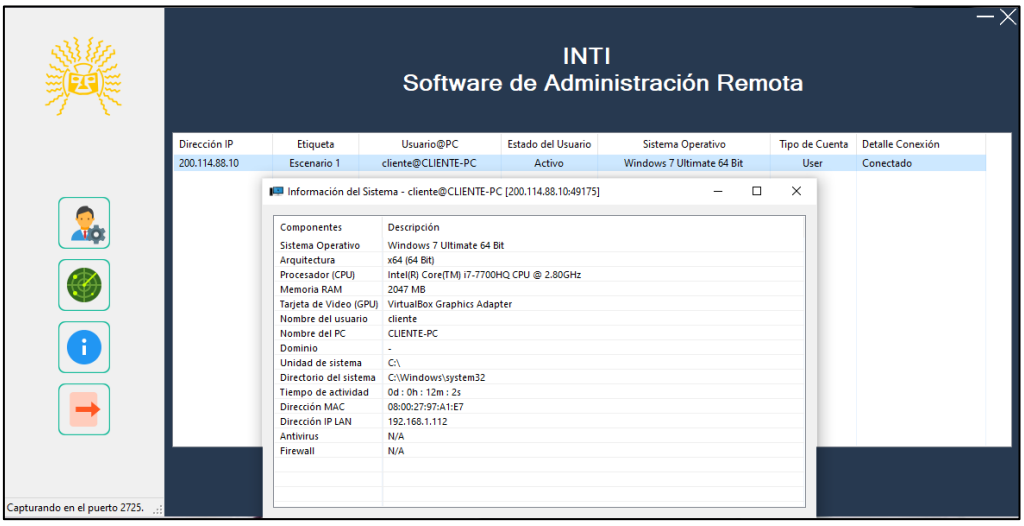


*Ilustración 38 Información sistema del cliente.*

Como se puede observar en la Ilustración 39 Escritorio remoto del cliente. se muestra la captura del escritorio de la víctima con Windows 7.

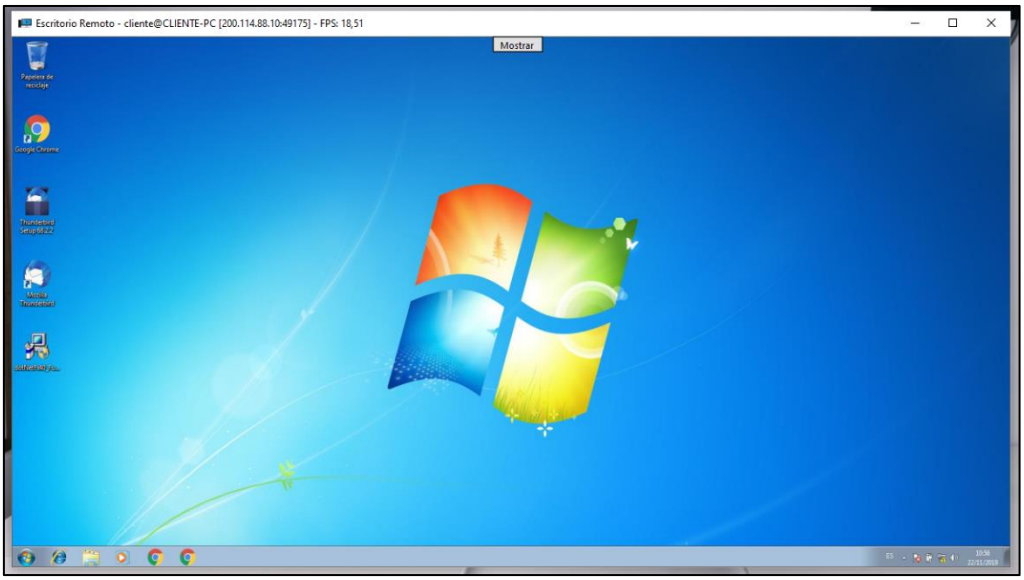


*Ilustración 39 Escritorio remoto del cliente.*

### 5.4.2. Resultado del Escenario 2

El resultado de conexión fue exitoso para ambos casos, en Windows 7 no existió ningún inconveniente para lograr la comunicación, mientras que para Windows 10 tuvimos que utilizar Cobra Crypter para ofuscar el Malware y pase desapercibido por el filtro de detección de amenazas de Windows Defender que viene integrado al sistema.

- Como se ve en la Ilustración 40, se produjo una conexión exitosa con la víctima con sistema operativo Windows 7.

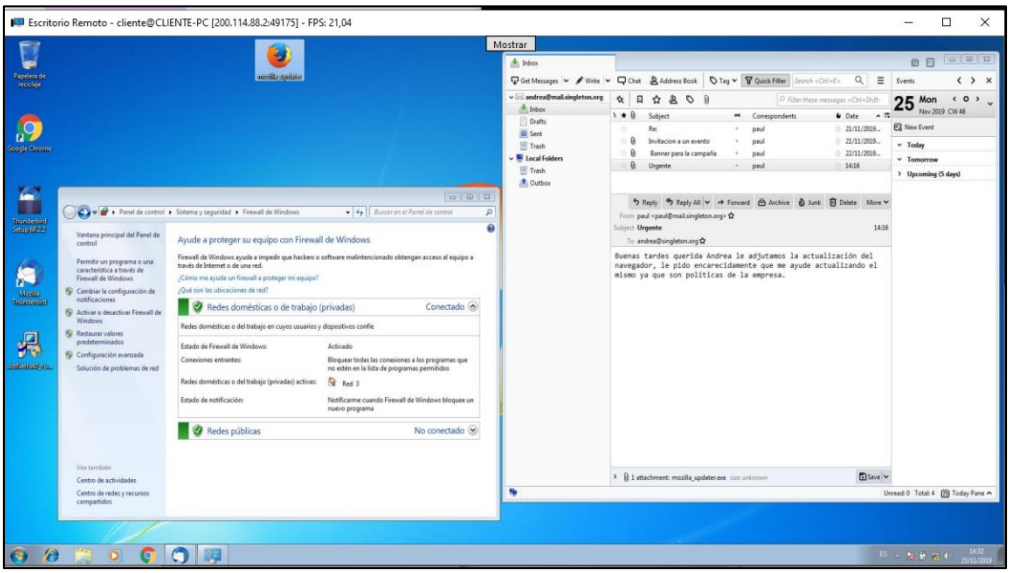


*Ilustración 40 Conexión con Windows 7.*

- De igual manera, hemos tomado el control de víctimas con el sistema operativo Windows 10, como se ve en la Ilustración 41.

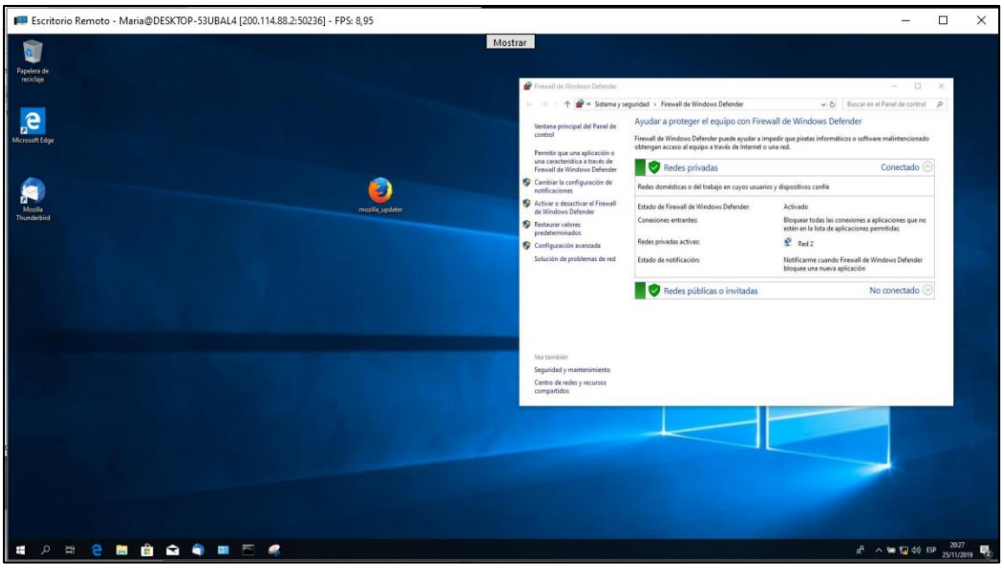


*Ilustración 41 Conexión con Windows 10.*

### 5.4.3. Resultado del Escenario 3

Para las pruebas de este escenario se utilizó McAfee Antivirus Plus, al momento de realizar un análisis al archivo este no lo detecto como Malware así que se pudo ejecutar sin inconvenientes, en el momento de la ejecución el antivirus tampoco lanzó ninguna alerta pasando totalmente inadvertido provocando que este se conecte con el servidor, como se puede observar en la Ilustración 42.

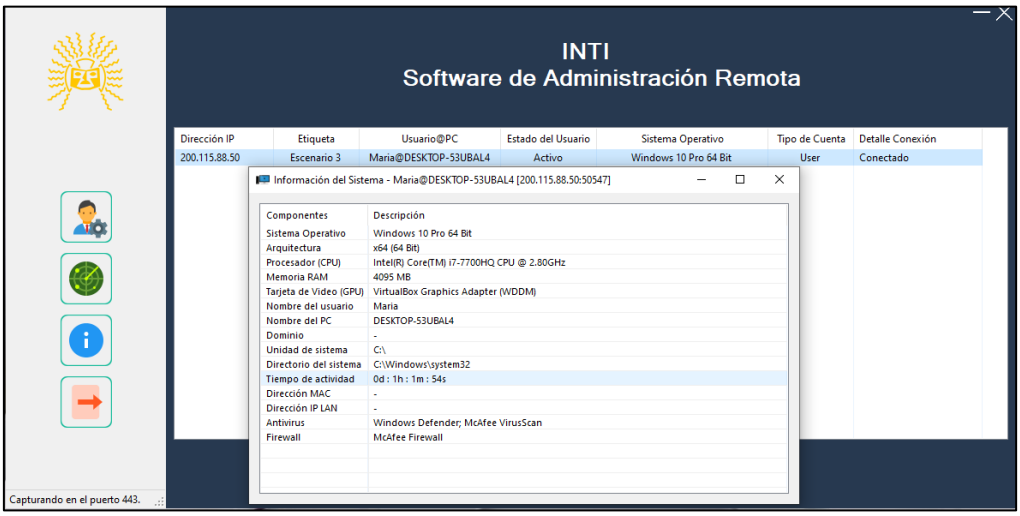


*Ilustración 42 Conexión con el Cliente con Antivirus.*

Por último, como se ve en la Ilustración 43, se tomó el control del escritorio de la víctima con Windows 10 en el cual se puede ver claramente que el antivirus se encuentra activo.

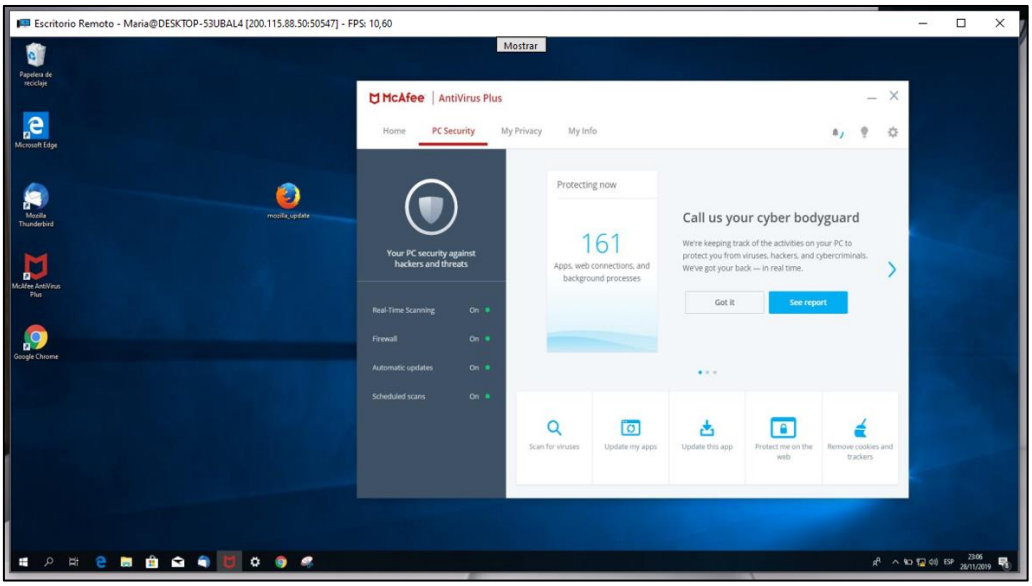


*Ilustración 43 Captura del escritorio remoto de la víctima.*

# 6. Conclusiones

En la presente tesis realizamos una investigación para el diseño, desarrollo, implementación y fases de prueba de un R.A.T. con la finalidad de evaluar la seguridad del sistema operativo Windows y los ataques que se encuentran especialmente a nivel de los sistemas troyanos. El R.A.T. que nosotros desarrollamos trabaja bajo la arquitectura cliente-servidor el cual permite tomar control absoluto del computador infectado lo que se convierte en una enorme falla de seguridad para los usuarios finales.

En nuestra investigación hemos escogido Windows de Microsoft como sistema operativo objetivo de análisis y ataque, debido a que es este es el dominante dentro de las empresas y dentro de los hogares, lo que lo convierte en un objetivo para los atacantes quienes se aprovechan de las vulnerabilidades que existen como falta de actualizaciones, Firewall, antivirus, etc., que son utilizadas para infringir la confidencialidad, disponibilidad, integridad y consistencia del sistema, en el cual el R.A.T. ataca al control de acceso del sistema operativo obteniendo el control total de la víctima.

En este contexto, durante el despliegue de los escenarios pudimos demostrar de manera práctica que un R.A.T. es capaz de evadir ciertas medidas de seguridad como son Firewalls y antivirus a través del uso de sistemas de ofuscación y Phishing, debido a que una gran cantidad de empresas no tienen control de los archivos adjuntos que llega por correo electrónico creando una brecha de seguridad que podemos aprovechar para que de esta manera la victima descargue nuestro Malware, lo ejecute y así tomar el control absoluto de su sistema.

De igual forma, hemos utilizado algunas técnicas de ingeniería social que facilita a los hackers la obtención de datos privilegiados de una entidad, debido a que estos persuaden al usuario con correos falsos en donde se suelen hacer pasar por una entidad legítima, anunciando que se han ganado un premio o, haciéndose pasar por un empleado de mayor rango en la empresa, en donde a través de un correo electrónico advierten al usuario la necesidad de descargarse un archivo adjunto, para “obligarlo” a recatar las ordenes que el hacker describe en su mensaje y así lograr el objetivo de infectar a la víctima.

Por lo tanto, un R.A.T. como mecanismo de administración remota en comparación con otros tipos de Malware que hacen funciones similares, este tiene un consumo mínimo de recursos del equipo infectado, pasando totalmente inadvertido gracias a que no posee integrado ningún mecanismo de recolección de información como Keyloggers, siendo esto muy beneficioso ya que no provoca lentitud en el sistema operativo y no llega a generar sospechas por parte de la víctima.

Finalmente, después de concluir nuestro trabajo de investigación, hemos determinado que el usuario es el “eslabón” más débil de la cadena de seguridad ya que como comprobamos en el desarrollo de nuestra tesis, implementamos escenarios con diferentes mecanismos de seguridad en los que se han aplicado políticas de Firewall y antivirus. Sin embargo, estos fueron evadidos mediante métodos de ofuscación y Phishing, logrando llegar a los usuarios para que estos descarguen y ejecuten nuestro R.A.T., en la que hemos determinado que existe un riesgo alto de desinformación en cada usuario, razón por la cual se necesita la implementación de campañas o políticas sobre las amenazas que podemos encontrar en Internet y los riesgos que pueden llegar a causar dentro de las empresas, con la finalidad de que los usuarios se concienticen y no abran archivos de orígenes desconocidos.

# 7. Referencias


[1] M. Soriano, «Seguridad en redes y seguridad de la información,» 2014. [En línea]. Available: http://improvet. cvut. cz/project/download/C2ES/Seguridad_de_Red_e_Informacion. pdf..

[2] R. Lemus, «Seguridad de la Información,» *Seguridad de la Información*, vol. 1, p. 160, 2014.

[3] K. C. Laundon y J. P. Laudon, Management Information Systems, Pearson Education Limited, 2014.

[4] G. Escrivá, R. Romero, D. Ramada y R. Pérez, Seguridad Informática, Madrid: Macmillan Iberia, 2013.

[5] M. E. Whitman y H. J. Mattord, Principles of Information Security, Boston: Course Technology Cengage Learning, 2012.

[6] F. Masahiro, J. Christian, A. Shiori, I. Yuki y N. Masakatsu, «Physical trust-based persistent authentication.,» de *13a Conferencia Anual de 2015 sobre Privacidad, Seguridad y Confianza (PST)*, Izmir, 2015.

[7] J. Costas, Seguridad y alta disponibilidad, Madrid: RA-MA Editorial, 2014.

[8] J. Niekerk y R. Solms, «From information security to cyber security,» *Computers & Security*, nº 38, pp. 97-102, 2013.

[9] R. Anderson, «Why information security is hard - an economic perspective,» *Seventeenth Annual Computer Security Applications Conference*, 2001.

[10] Iso27000.es, «El portal de ISO 27001 en Español,» 2020. [En línea]. Available: http://www.iso27000.es/sgsi.html.

[11] Y. Cheng, W. Wang, J. Wang y H. Wang, «FPC: A new approach to Firewall policies compression,» *Tsinghua Science and Technology*, pp. 65 - 76, 2018.

[12] J. P. Esparza, «Implementación de un Firewall sobre plataforma linux en la empresa de contabilidad armas y asociados,» Enero 2013. [En línea]. Available: https://bibdigital.epn.edu.ec/bitstream/15000/6056/1/CD-4785.pdf.

[13] M. Buddhikot, A. Hari, S. Kundan y M. Scott, «MobileNAT: A New Technique for Mobility Across Heterogeneous Address Spaces,» *Mobile Networks and Applications, Springer Nature B.V.*, nº 3, pp. 298-302, 2005.

[14] P. Kabala y D. Laskowski, «Analysis of Network Traffic Filtering / Analiza Filtracji Ruchu Sieciowego,» *Journal of KONBiN*, nº 1, pp. 41-60, 2015.

[15] C. Jake y L. Jun, «Un prototipo basado en OpenFlow de Firewalls de hardware con estado orientados a SDN,» de *22a Conferencia Internacional IEEE 2014 sobre protocolos de red*, Raleigh, 2014.

[16] K. Swati Maloki y M. Sudhakar, «Design and Implementation of Hardware Firewall Using FPGA,» *2018 3rd International Conference for Convergence in Technology (I2CT)*, pp. 1-4, 2018.

[17] Z. Trabelsi y V. Molvizadah, «Edu-Firewall device: An advanced Firewall hardware device for information security education,» *2016 13th IEEE Annual Consumer Communications & Networking Conference (CCNC)*, pp. 278-279, 2016.

[18] X. Yue, W. Chen y Y. Wang, «The research of Firewall technology in computer network security,» *2009 Asia-Pacific Conference on Computational Intelligence and Industrial Applications (PACIIA)*, pp. 421-424, 2009.

[19] K. Dadheech, A. Choudhary y G. B. Bhatia, «De-Militarized Zone: A Next Level to Network Security,» *2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)*, pp. 595-600, 2018.

[20] D. Mukul, G. Himanshu, M. Somya y B. B, «Implementación de caché utilizando

inteligencia colectiva en la arquitectura antivirus basada en la nube,» *Conferencia Internacional 2016 sobre Control Avanzado de Comunicación y Tecnologías de Computación (ICACCCT),* pp. 593 -595, 2016.

[21] Á. P. Guijarro, Seguridad perimetral, 2012.

[22] E. Maiwald, Network Security A Beginner's Guide, New York: McGraw-Hill , 2013.

[23] N. A. Hassan, Digital Forensics Basics, New York, New York, USA: Apress, 2019.

[24] E. Quero, Sistemas operativos y lenguajes de programación, Editorial Paraninfo, 2002.

[25] S. Subramanya y N. Lakshminarasimhan, «Computer viruses,» *IEEE Potentials,* pp. 16-19, 2001.

[26] A. Bettany y M. Halsey, Windows Virus and Malware Troubleshooting, New York: Apress Media, 2017.

[27] H. Nihad y H. Rami, Digital Privacy and Security Using Windows, New York: Apress, 2017.

[28] Instituto Nacional de Cyber Seguridad, «incibe,» [En línea]. Available: https://www.incibe.es/sites/default/files/contenidos/guias/doc/guia_glosario_ciberseguridad_metad.pdf. [Último acceso: 20 Agosto 2019].

[29] M. Sikorski y A. Honing, Practical Malware Analysis, San Francisco, CA: No Starch Press, 2012.

[30] N. Reddy, Practical Cyber Foresincs, Apress, 2019.

[31] M. Sikorski y A. Honig, MALWARE ANALYSIS: The Hands-On Guide to, San Francisco: No Starch Press, 2012.

[32] C. Hadnagy, Social engineering: The art of human hacking, Indianapolis: Wiley Publishing, 2011.

[33] I. Mann, Hacking the Human Social Engineering Techniques and, 2008.

[34] K. Mitnick y W. Simon, Controlling the Human Element of Security, New York: John Wiley & Sons, 2002.

[35] E. Medina, «Hacking Ético: Una herramienta para la seguridad informática,» *Tesis de Licenciatura. Universidad Piloto de Colombia.,* 2015.

[36] A. R. Plazas Garcia, «Ingeniería social en las empresas colombianas,» 23 04 2018. [En línea]. Available: https://repository.unad.edu.co/handle/10596/18704.

[37] E. Y. Rodríguez, Metodologías de ingeniería social, Universitat Oberta de Catalunya, 2018.

[38] C. E. Lopéz Grande y R. Salvador Guadrón, Ingeniería Social: El Ataque Silencioso, ITCA, Editores, 2015.

[39] M. Beron, P. Henriques, M. Varanda y R. Uzal, «Herramientas para la comprensión de programas,» Junio 2006. [En línea]. Available: https://docplayer.es/35195529-Herramientas-para-la-comprension-de-programas.html.

[40] L. C. Miller, Modern Malware For Dummies, New Jersey, 2012.

[41] C. Barria, D. Cordero, C. Cubillos y R. Osses, «Obfuscation procedure based in dead code insertion into crypter,» *2016 6th International Conference on Computers Communications and Control (ICCCC),* pp. 23-29, 2016.

[42] C. Barria, D. Cordero, C. Cubillos, H. Allende y C. Casado, «Obfuscation procedure based on the insertion of the dead code in the crypter by binary search,» *2018 7th International Conference on Computers Communications and Control (ICCCC),* pp. 183-192, 2018.

[43] D. Master, Introducción a la Esteganografía, 2004.

[44] D. Lerch y D. Megías, «Esteganografía en zonas ruidosas de la imagen,» 2014. [En línea]. Available: http://hdl.handle.net/10045/40424.

[45] K. R. Sudhir, M. K. Jha, S. Laxmi, N. Rahul, J. Abhishek y S. Sutapa, «Quantum

cryptography for IoT: APerspective,» *2017 International Conference on IoT and Application (ICIOT),* pp. 1-4, 2017.

[46] K. İlker y A. Murat, «The Ghost in the System: Technical Analysis,» *International Journal on Information Technologies & Security,* vol. 11, pp. 73-84, 2019.

[47] M. Marchetti, F. Pierazzi y M. Colajanni, «A. Analysis of high volumes of network traffic for Advanced Persistent Threat detection.,» *Computer Networks,* vol. 109, pp. 127-141, 2016.

[48] S. Samuel, J. Graham y C. Hinds, «Hunting Malware: An Example Using Ghost,» *IEEE,* pp. 97-102, 2017.

[49] M. Yamada, M. Morinaga, Y. Unno y M. Takenaka, «RAT-based malicious activities detection on enterprise internal networks,» *IEEE,* pp. 321-325, 2015.

[50] M. A. Maarof y S. Z. Mohd Shaid, «Malware behavior image for Malware variant identification,» *IEEE,* pp. 238-243, 2014.

[51] K. Tangen, «Detecting Remote Administration Trojans through Dynamic Analysis using Finite-State Machines.,» *Semantic Scholar,* 2014.

[52] T. Shigemoto, S. Fujii , I. Kuriama, T. Kito, H. Nakakoji, Y. Fujii y H. Kikuchi, «Development of White List Based Autonomous Evolution of Defense System for RAT Malware,» *IEEE,* pp. 95-101, 2018.

[53] Microsoft Corporation, «Get started with the .NET Framework,» 04 01 2019. [En línea]. Available: https://docs.microsoft.com/en-us/dotnet/framework/get-started/index.

[54] A. Aguirre, L. Rodas y I. Nikita, «ANÀLISIS, DISEÑO, DESARROLLO E IMPLEMENTACIÓN,» Mayo 2016. [En línea]. Available: https://repositorio.espe.edu.ec/bitstream/21000/11961/1/T-ESPE-053263.pdf.

[55] N. Gomez, «HEADSEM,» 15 Septiembre 2017. [En línea]. Available: https://www.headsem.com/por-que-podria-ser-c-el-lenguaje-de-programacion-indicado-para-ti/.

# 8. Anexos

## 8.1. Diagrama de Estados

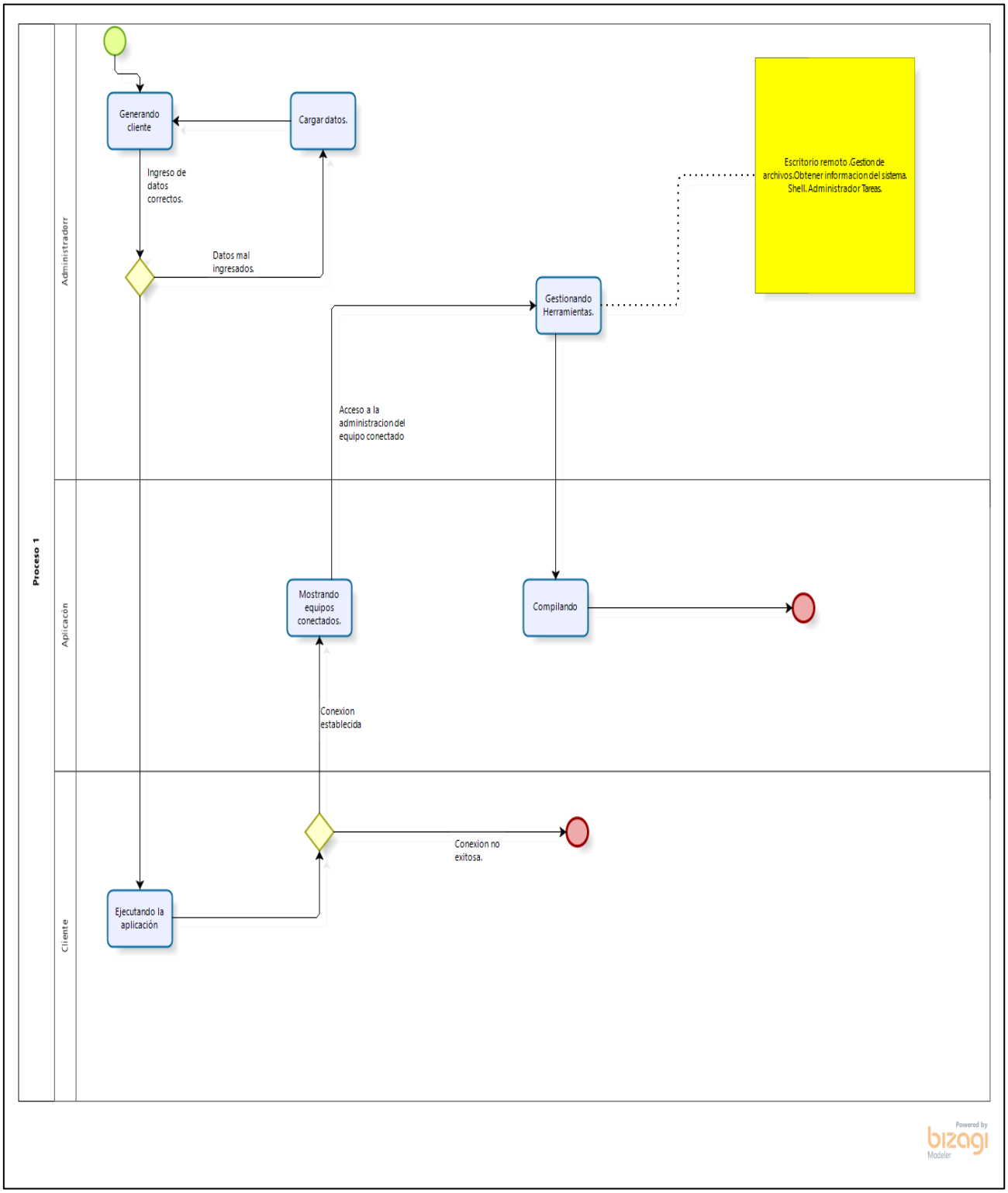


*Ilustración 44 Diagrama de estado.*

## 8.2. Manual de Usuario

Requisito previo que debe tener instalado la máquina que va a actuar como servidor.

- Instalar .NET Framework 4.0 en el servidor y en la victima

Como se ve en la Ilustración 45, estos son todos los archivos necesarios para desplegar el R.A.T.

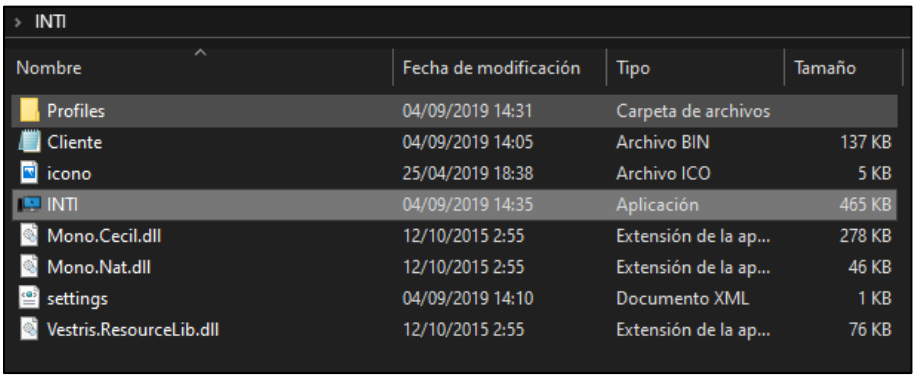


*Ilustración 45 Archivos del servidor.*

Ejecute el archivo INTI.exe. Si por algún motivo no se ejecuta es recomendable agregar el ejecutable a la lista de exclusiones de nuestro antivirus, como muestra en la Ilustración 46 con McAfee Antivirus.

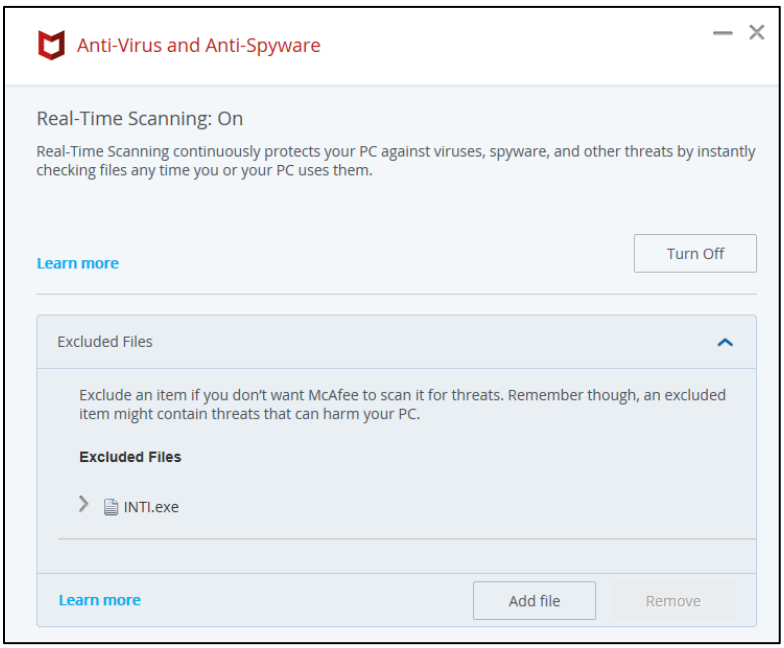


*Ilustración 46 McAfee Antivirus.*

En el caso de que no sea detectado ni eliminado como Malware, el programa empezara a ejecutarse con normalidad. Una vez se termine de ejecutar se mostrará la ventana principal, como se puede observar en la Ilustración 47.

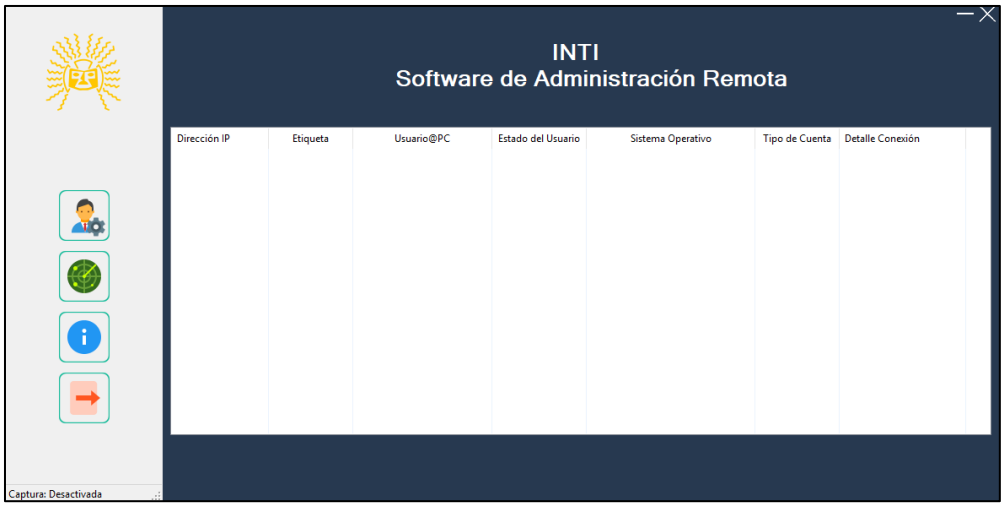


*Ilustración 47 Ventana Principal de INTI R.A.T.*

- Generar Cliente

Lo primero que debemos hacer es generar el cliente para la conexión entre el cliente y servidor. Para configurar este cliente damos clic sobre al primer icono de la izquierda. Dentro de esta ventana debemos especificar una etiqueta(nombre) a la conexión, dirección IP (dirección privada o pública) de nuestra red junto con un puerto, como se ve en la Ilustración 48. También debemos establecer una contraseña, esta contraseña debe ser la misma que debemos utilizar en el momento de capturar los clientes.

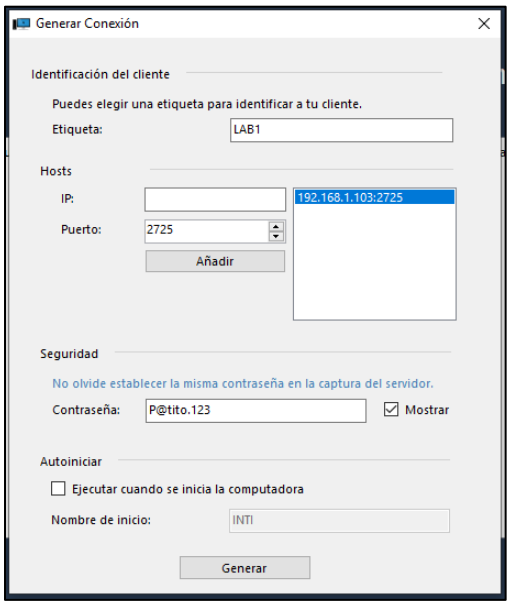


*Ilustración 48 Ventana generar Cliente.*

Si los datos ingresados son correctos, se creará un archivo .exe en la ruta especificada, como se ve en la Ilustración 49.

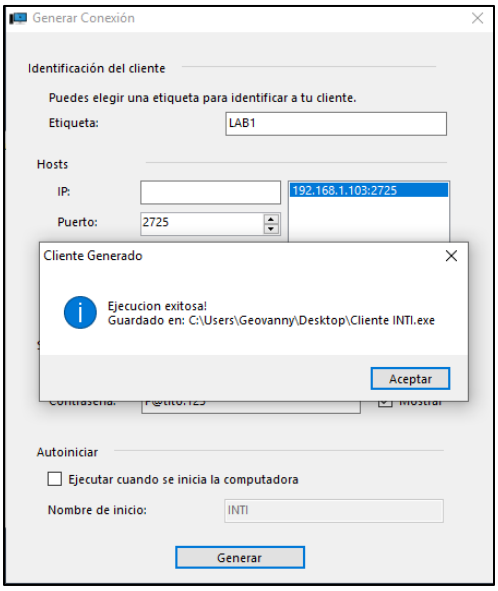


*Ilustración 49 Creación del cliente.*

- Análisis del Cliente Generado

Existen varios programas online que realizan el análisis de archivos maliciosos, en nuestro caso decidimos utilizar Virus Total el cual nos dio los siguientes resultados.

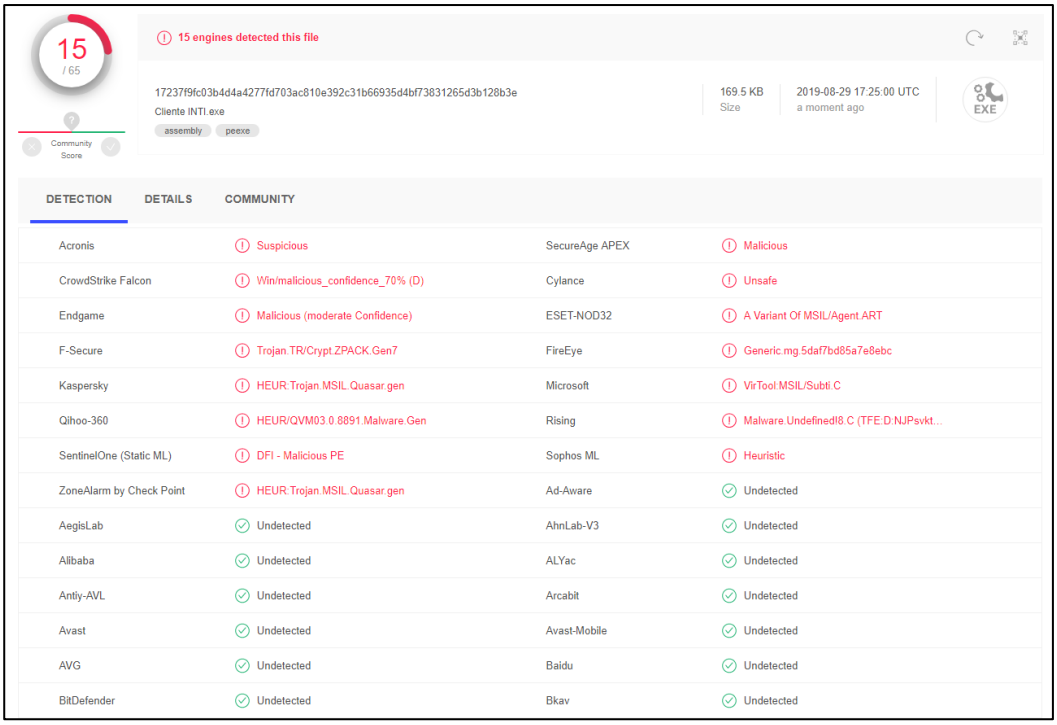


*Ilustración 50 Resultado de virus total.*

Como se puede ver en la Ilustración 50, una gran cantidad de antivirus aun detectan nuestro R.A.T es por ello que vamos a utilizar una técnica de ofuscación para evitar esto.

- Ofuscación del Cliente

Para evitar que ciertos antivirus detecten nuestro R.A.T. utilizamos un software gratuito denominado Cobra Crypter 2019. Existen otras variantes de Crypter`s de pago

diseñados para hacer que un Malware pase totalmente inadvertido ante cualquier antivirus.

Como se puede ver en Ilustración 51, en la pantalla principal de Cobra Crypter debemos seleccionar el archivo .exe y un icono en formato .ico.

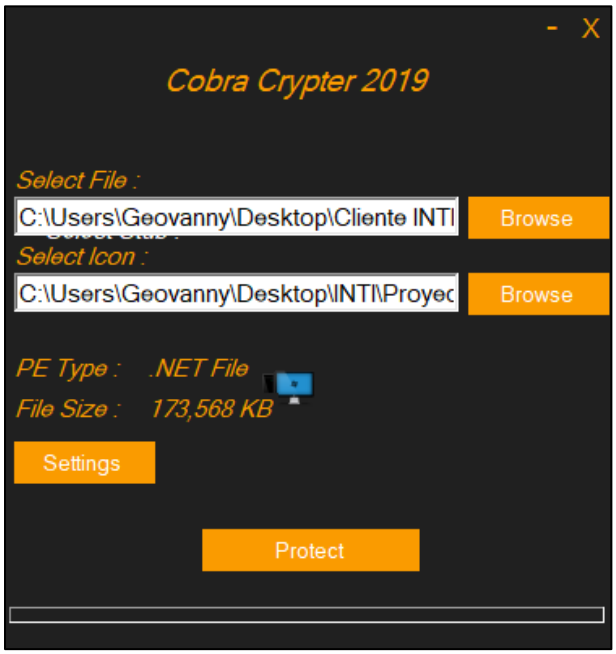


*Ilustración 51 Cobra Crypter.*

Luego ingresamos en Settings, como se ve en la Ilustración 52, en la pestaña opciones debemos seleccionar las casillas unnecessary codes y obfuscator, también debemos seleccionar la versión de .NET Framework con la que está desarrollada la aplicación en nuestro caso es la versión 4.0.

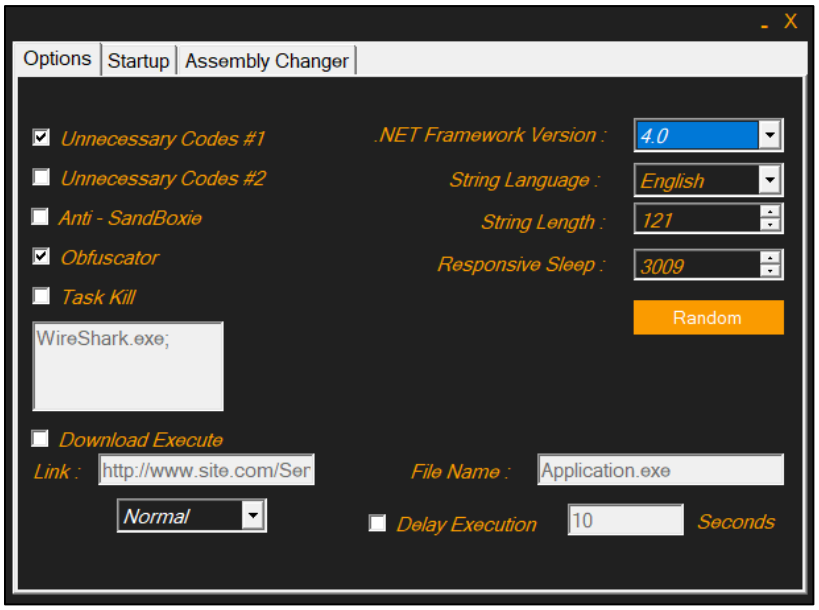


*Ilustración 52 Ventana de creación de archivo ofuscado.*

Como se puede observar en la Ilustración 53, en la ventana de Assembly Changer podremos especificar la información de ensamblado para el ejecutable o seleccionar uno

de los ya existentes como se muestra en la siguiente ilustración.

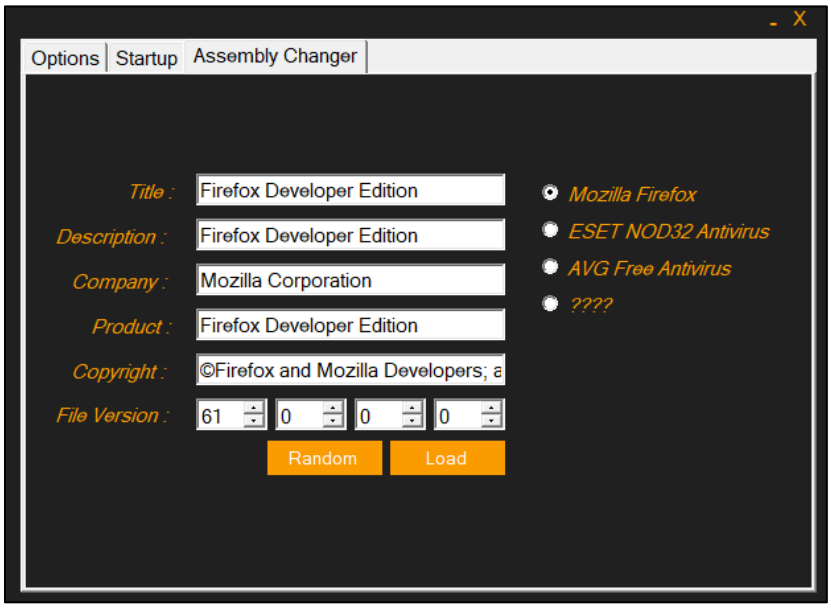


*Ilustración 53 Ventana de elección de icono con el que se guardara.*

Teniendo todo lo anterior configurado seleccionamos Protect, esperamos unos segundos hasta que se genere el nuevo ejecutable.

Finalmente, como se ve en la Ilustración 54, volvemos a analizar en Virus Total obteniendo como resultado que Windows Defender, McAfee, Panda, entre otros antivirus ya no lo reconocen.

| | | | |
|---|---|---|---|
| MaxSecure | Undetected | McAfee | Undetected |
| Microsoft | Undetected | NANO-Antivirus | Undetected |
| Palo Alto Networks | Undetected | Panda | Undetected |
| Rising | Undetected | Sophos AV | Undetected |
| SUPERAntiSpyware | Undetected | TACHYON | Undetected |
| Tencent | Undetected | Trapmine | Undetected |
| TrendMicro | Undetected | TrendMicro-HouseCall | Undetected |
| ViRobot | Undetected | Webroot | Undetected |
| Yandex | Undetected | Zillya | Undetected |

*Ilustración 54 Resultado del análisis de Malware en virus total después de ofuscar.*

- Análisis del R.A.T. ya ofuscado

Antes de enviar a la víctima verificamos que McAfee no detecte nuestro R.A.T como un Malware. Los resultados del análisis del Malware se muestran en la Ilustración 55.

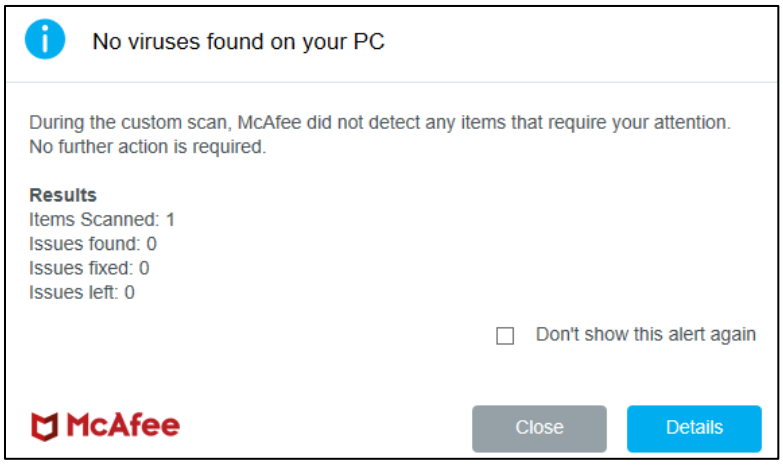


*Ilustración 55 Resultado del análisis de McAfee.*

Una vez que tenemos listo el archivo .exe mediante Ingeniería Social enviamos el ejecutable a las víctimas.

INTI soporta las versiones de Windows 7, Windows 8.1 y Windows 10. En la Ilustración 56, Ilustración 57 y Ilustración 58 se muestra que nuestro prototipo se encuentra ejecutando en cada uno de estas versiones.

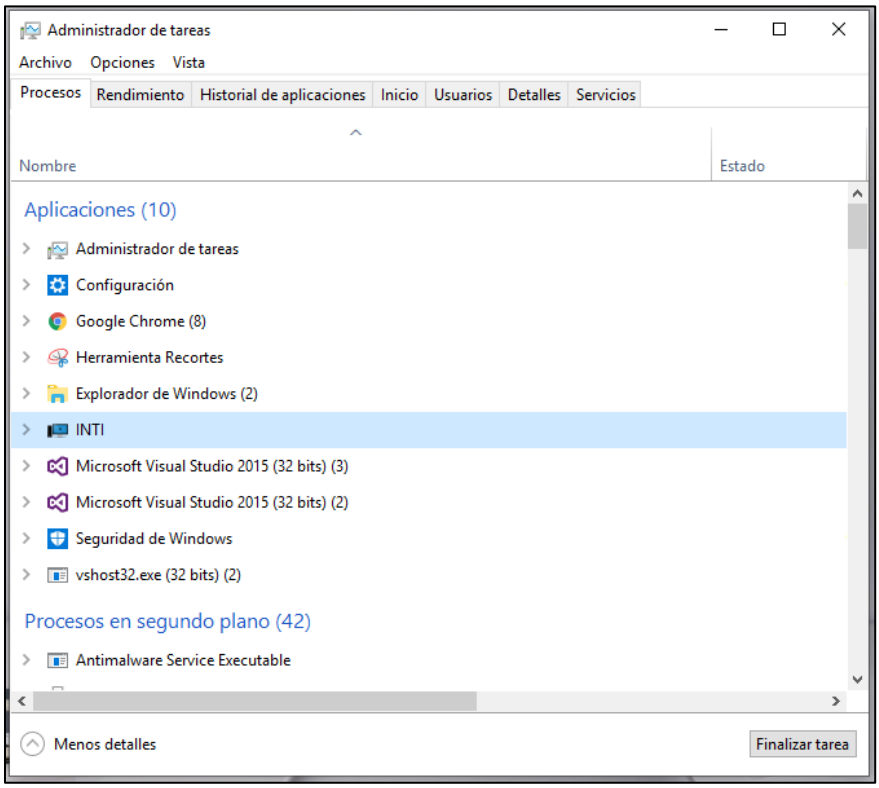


*Ilustración 56 Administrador de tareas de Windows 10.*

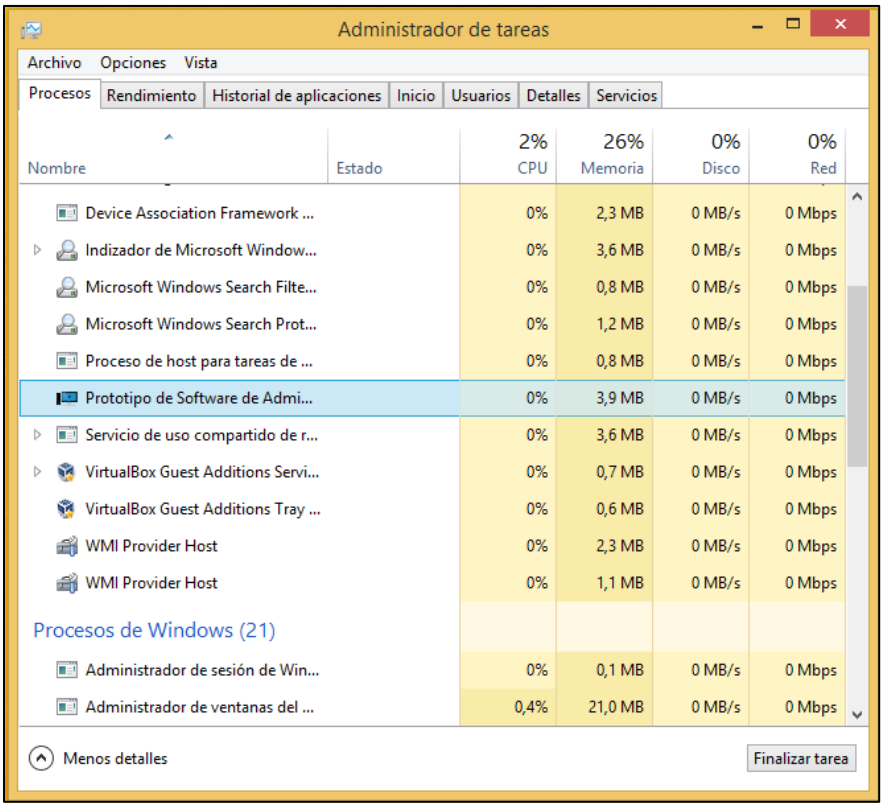


*Ilustración 57 Administrador de Tareas de Windows 8.1.*

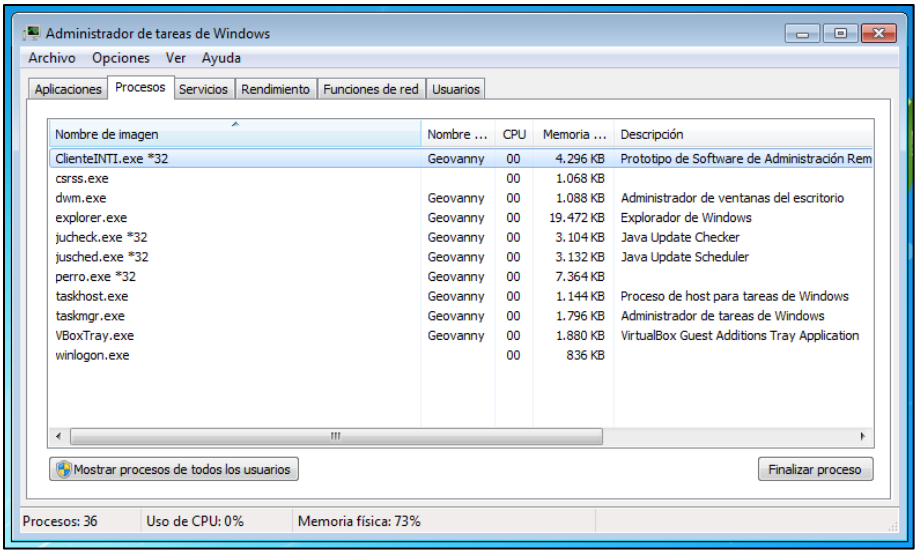


*Ilustración 58 Administrador tarea de Windows 7.*

- Captura de los equipos de las Victimas

Regresamos a la ventana principal de la aplicación, seleccionamos el segundo icono de la izquierda el cual nos abrirá la ventana Capturar donde debemos ingresar el mismo puerto y contraseña que ingresamos para crear el Cliente, como podemos ver en la Ilustración 59. Podemos marcar los campos buscar conexiones al iniciar y mostrar notificaciones de nuevas conexiones si desea (campos opcionales).

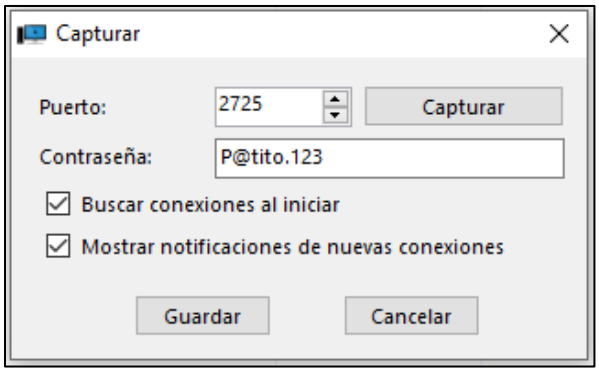


*Ilustración 59 Habilitar captura de clientes por puertos.*

Si marco la casilla mostrar notificaciones de nuevas conexiones se notificará cada que un nuevo cliente se conecte al servidor, como se muestra en Ilustración 60.

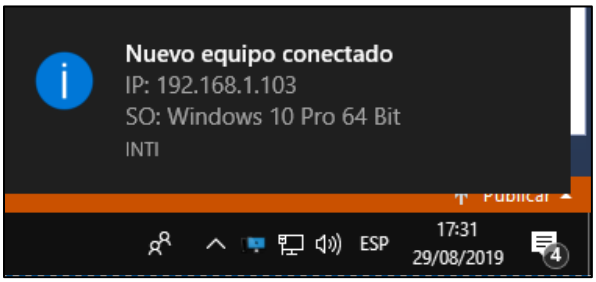


*Ilustración 60 Notificación de equipo conectado.*

- Lista de equipos Conectados


*Ilustración 61 Clientes conectados.*

Cuando damos clic derecho sobre un cliente podremos acceder a dos opciones: las opciones de conexión y las herramientas; dentro de las opciones de conexión podremos reconectar y eliminar el cliente en el caso de que se tenga problemas, ver Ilustración 62.

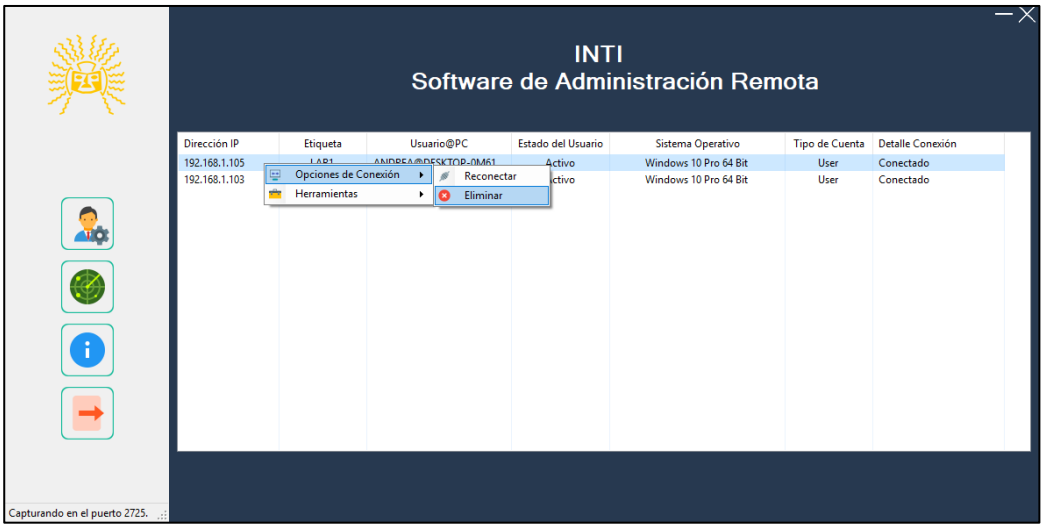


*Ilustración 62 Opciones del cliente conectado.*

Como se ve en la Ilustración 63, dentro de las Herramientas tenemos acceso a: Información del sistema, Administración de archivos, Administrador de tareas, Consola remota, Escritorio Remoto.

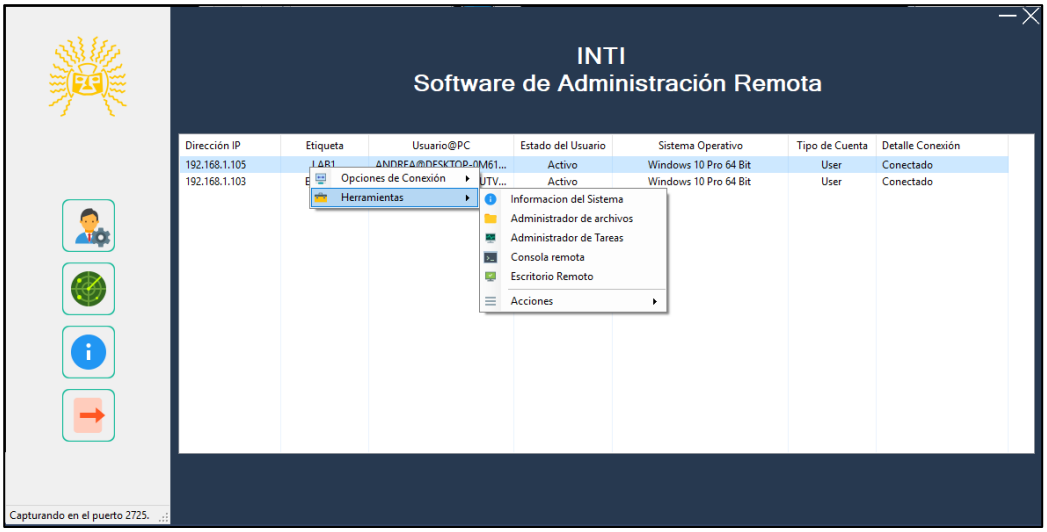


*Ilustración 63 Herramientas de clientes conectados.*

También contamos con un apartado de Acciones en donde tenemos accesos directo a las funciones de apagado, reinicio y suspensión del equipo, como se puede observar en la Ilustración 64.

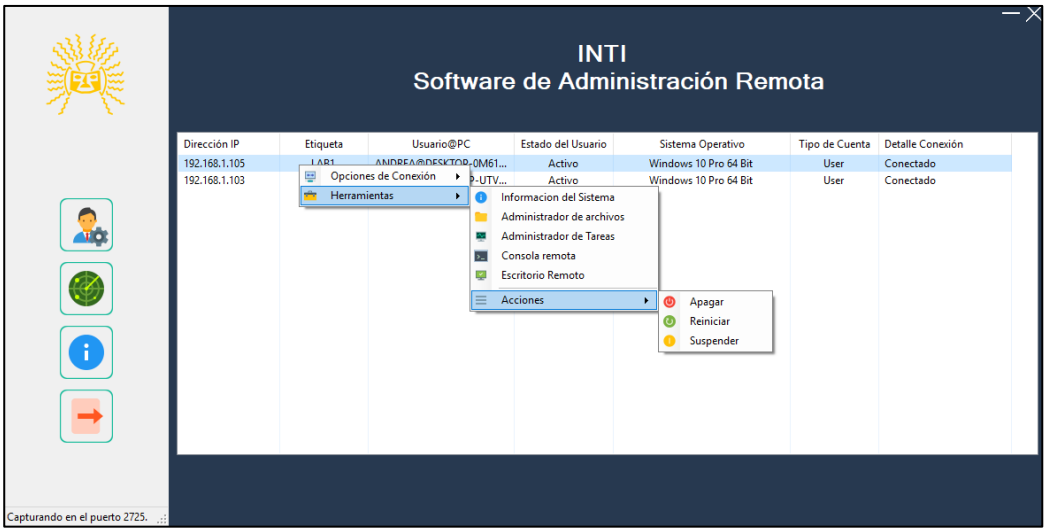


*Ilustración 64 Acciones de cliente conectado.*

Dentro de Información del Sistema, Ilustración 65, tenemos acceso a información del Sistema Operativo, Arquitectura, Procesador, Memoria RAM, Tarjeta de Video, Nombre de Usuario, Nombre del PC, Antivirus, Firewall, etc.

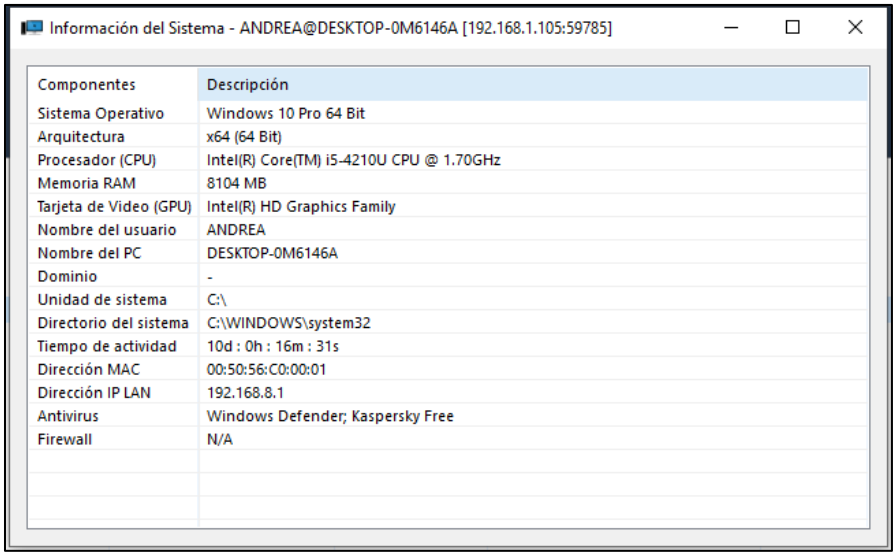


*Ilustración 65 Información de cliente conectado.*

Dentro de la administración de archivos, Ilustración 66, tenemos acceso a todos los discos con los que cuente el equipo del cliente. Se recomienda no acceder a la carpeta Windows. En esta ventana podremos tanto subir como bajar archivos, en el caso de bajarse algún tipo de archivo se creará una carpeta que almacenará estos archivos. También podemos mandar a ejecutar, renombrar y eliminar algún tipo de archivo.

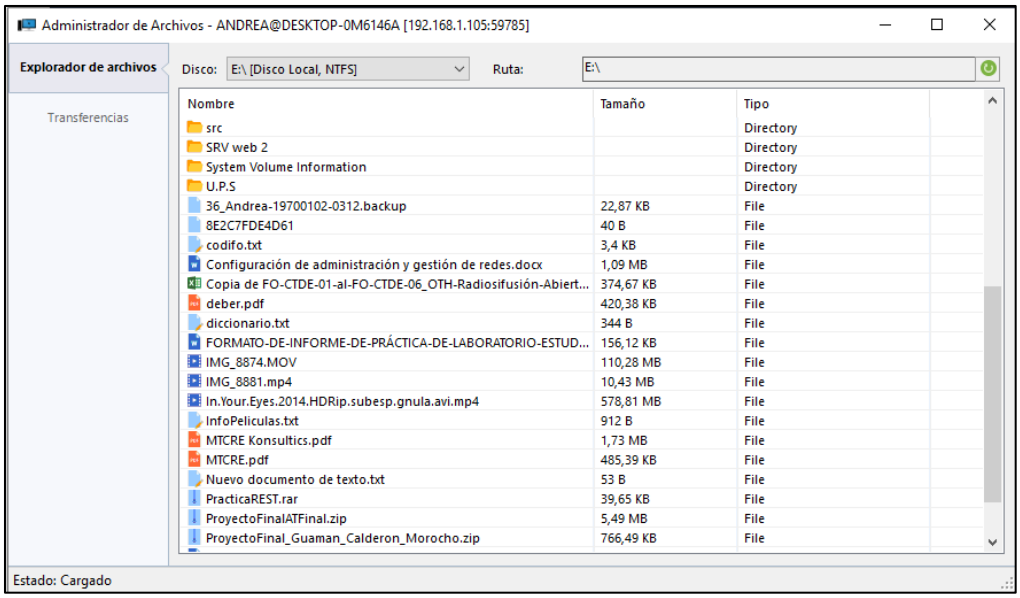


*Ilustración 66 Administración de archivos.*

En la ventana de Administración de Tareas, Ilustración 67, podemos finalizar los procesos que se estén ejecutando en el Sistema Operativo de la Victima.

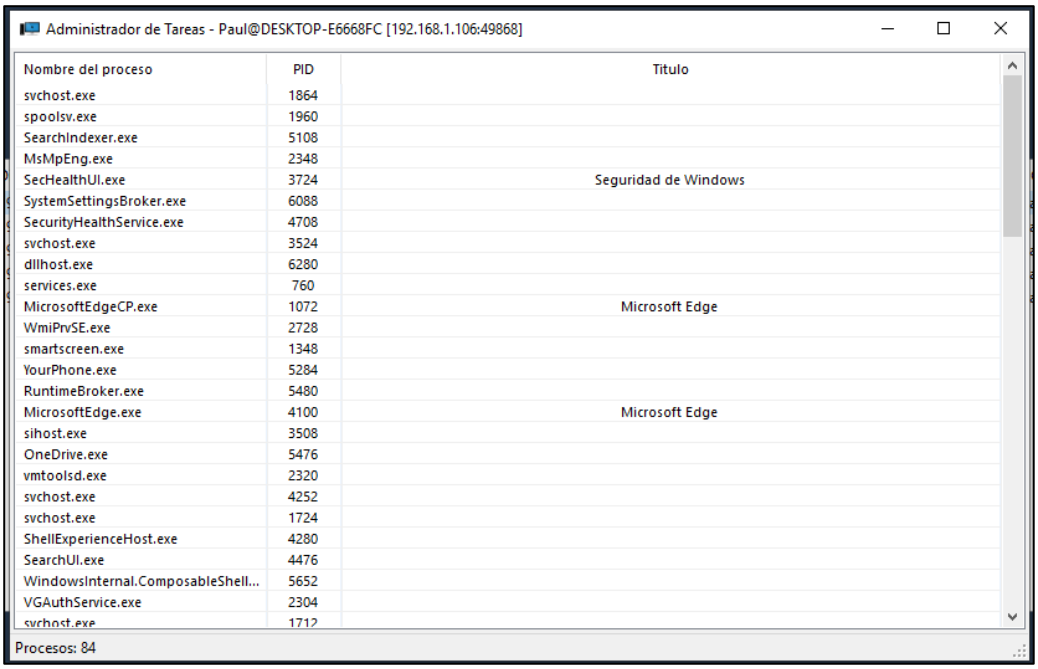


*Ilustración 67 Administrador de archivos equipo cliente.*

Dentro de la consola remota o Shell, Ilustración 68, podemos ejecutar todos los comandos admitidos por Windows como si se estuviese trabajando en el mismo equipo.

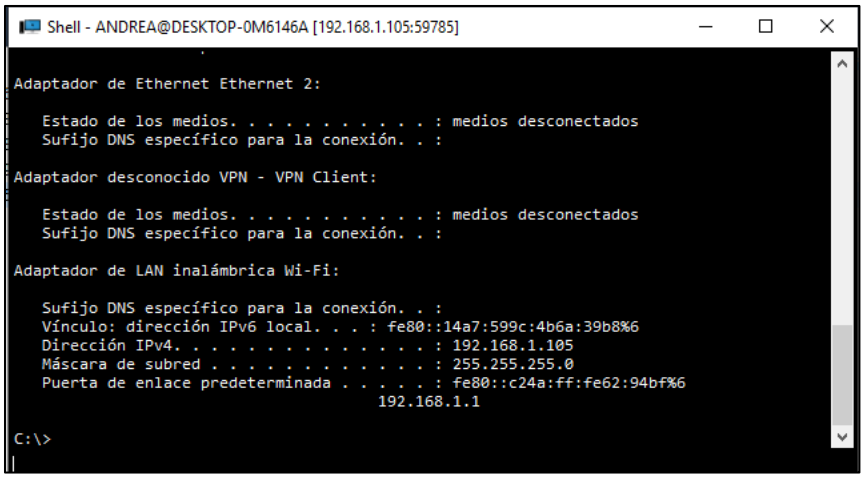


*Ilustración 68 Shell Cliente.*

Finalmente, como podemos observar en la Ilustración 69, Ilustración 70 e Ilustración 71 se muestra la funcionalidad de escritorio remoto en las diferentes versiones de Windows. Cabe recalcar que se puede habilitar y deshabilitar las funciones del mouse y teclado en el momento que requiera el usuario.

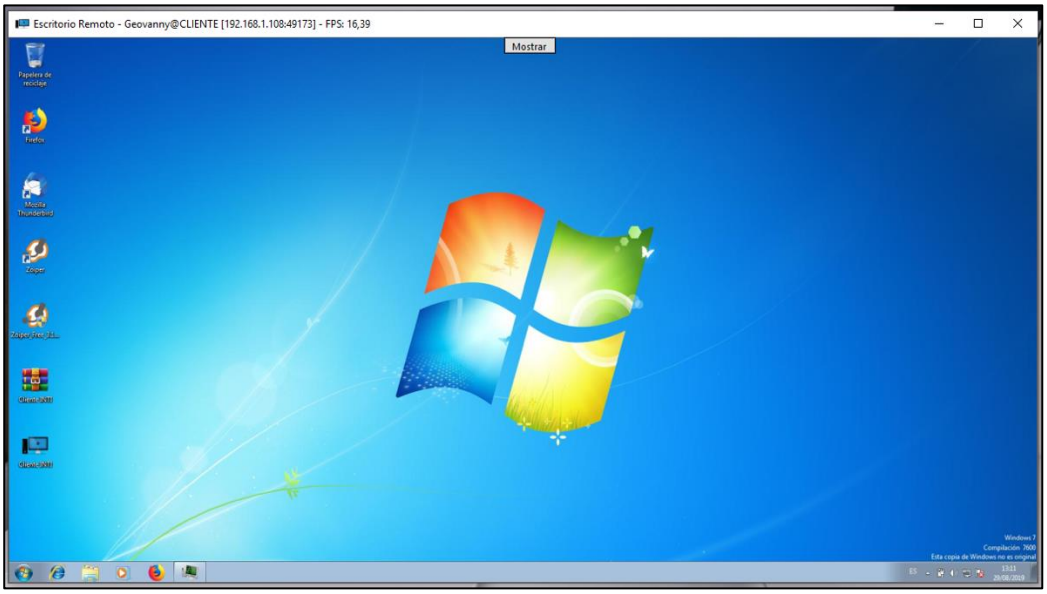


*Ilustración 69 Escritorio remoto de la víctima con Windows 7.*

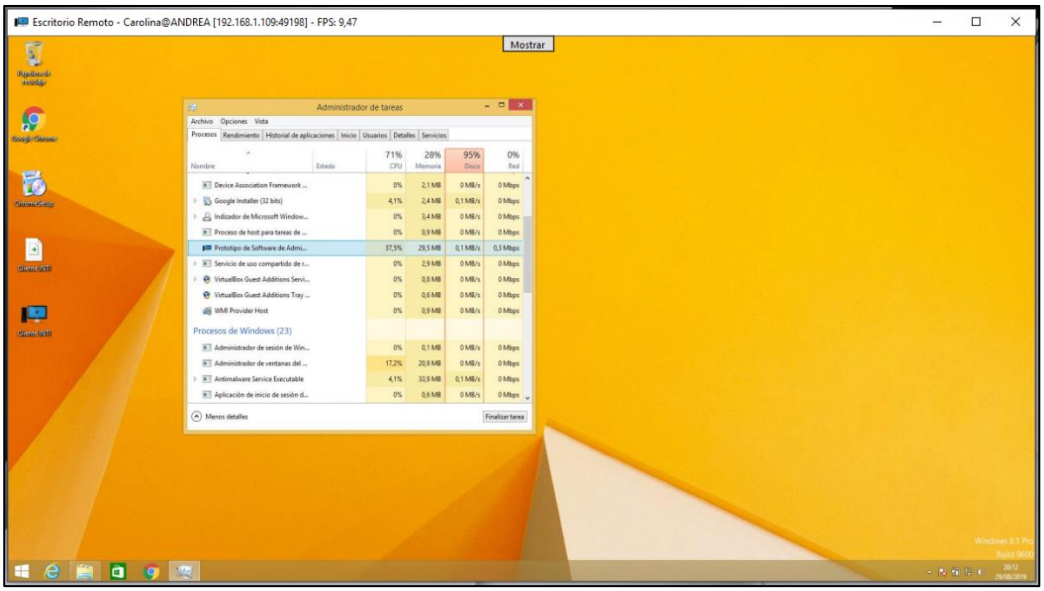


*Ilustración 70 Escritorio remoto de la víctima con Windows 8.1.*

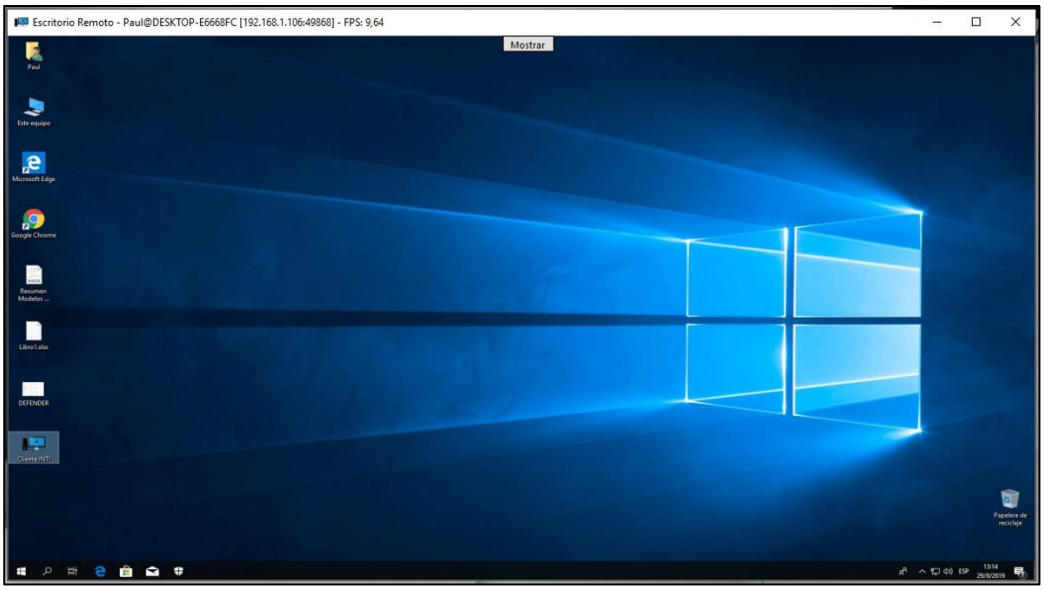


*Ilustración 71  Escritorio remoto de la víctima con Windows 10.*