Detección de canales encubiertos en la Capa de Red

Ignacio Martín Gallardo Urbini[12], Santiago Manuel Rudi[3]

1 Centro de Investigación y Desarrollo de Software Operacional del Ejército Argentino
2 Facultad de Ingeniería del Ejército - Universidad de la Defensa Nacional
3 Facultad de Ingeniería - Universidad de Palermo
4 Facultad de Ingeniería - Universidad Nacional de La Plata

Ignacio Martín Gallardo Urbini, igallardo@est.iue.edu.ar
Santiago Manuel Rudi, srudi@palermo.edu

Resumen. La presente investigación muestra el desarrollo e implementación de una técnica para detectar canales encubiertos en la capa de red del modelo OSI, por medio de la aplicación de algoritmos de machine learning. Junto con la definición del Protocolo de Internet, se realiza un análisis a fondo de aplicabilidad esteganográfica en el mismo, implicando posibles comunicaciones encubiertas entre un emisor y un receptor. Se abordan también tecnologías de control, análisis de tráfico y aprendizaje automático con el objetivo de detectar canales laterales de comunicación dentro del marco de Internet Protocol. Las aplicaciones de esta técnica quedarán asentadas con un caso de estudio para validar el modelo.

Palabras-clave: Esteganografía, Protocolo de Internet, Comunicaciones Encubiertas, Criptografía, Machine Learning.

1 Introducción

Eran mediados del siglo XIV, la dinastía Yuan reinaba el territorio chino, y los gobernantes, a fin de asegurar su dominio, imponían la orden de que cada diez familias usaran un solo cuchillo en su vida cotidiana (entre otras atrocidades y represiones) con el objetivo de que la gente quedase sin armas de metal por si querían levantarse en rebelión. Los habitantes hartos de las injusticias, decidieron sublevarse; para ello, los organizadores concibieron la idea de promover a los vecinos a regalarse mutuamente unas tortas llamadas "moon" en vísperas de las fiestas de otoño. Dentro de las tortas se puso una pequeña octavilla con un mensaje táctico para actuar en conjunto. La rebelión fue un éxito total y derivó en el establecimiento de la dinastía Ming.

El individuo, desde el momento en que experimentó la motivación de desenvolverse socialmente, sintió la necesidad de comunicarse, pero muchas veces, dicha comunicación debiera ser secreta para cierta porción de la sociedad; es decir, de alguna u otra forma debió comunicarse de manera oculta o encubierta con algunas personas, sin que otras se enteren de dicha interacción.

Eficientemente, los pobladores chinos pudieron comunicarse de manera incógnita para llevar a cabo la organización de la revuelta, logrando que dicha interacción pase desapercibida para los integrantes de la dinastía Yuan.

En la presente investigación, se abordará el concepto de *comunicación esteganográfica* como la ciencia de la *comunicación encubierta.* Se la llamará "ciencia", y no "arte", ya que este último pertenece a una condición humana personal que pudiendo o no involucrar conocimientos previos, tiene aptitud para generar hechos nuevos, pero cuando comienza a regirse por sus leyes, deja de ser arte para transformarse en ciencia.

El objetivo de este desarrollo apunta a realizar un análisis de aquellos campos en el encabezado IP donde es factible adicionar información de forma tal de crear un canal lateral, sin afectar la funcionalidad del protocolo.

Para comenzar, se presentará un breve marco teórico con el objetivo de incorporar las diferentes definiciones necesarias para comprender el desarrollo. No obstante, se presentarán conceptos sobre IPv4 (Internet Protocol Versión Cuatro), esteganografía, análisis de flujo de paquetes de datos en la red y machine learning.

Luego, para validar la propuesta, se llevará a cabo el diseño contextual de aplicación de un componente de software de comunicación encubiertas vía capa de red, otro módulo de software para análisis de flujos de datos y otro para aplicar algoritmos de machine learning con el fin de detectar canales laterales de comunicación.

Para finalizar, se mostrarán y compararán los resultados de las pruebas, se abordarán las conclusiones, líneas futuras de investigación y referencias bibliográficas utilizadas para la redacción del contenido de este documento.

2 Esteganografía y Criptografía

Se definirá entonces, *comunicación encubierta* o *esteganográfica,* al acto de comunicación entre dos o más actores, a través de un canal determinado de comunicación, sin que algún tercero con acceso a ese canal tenga conocimiento de dicha interacción. Se llamará *esteganografía,* a la disciplina que permite escribir en forma encubierta utilizando procedimientos científicos, y ocultando así mensajes u objetos dentro de otros llamados *portadores*, de modo que no se perciba su existencia.

Si bien la *esteganografía* suele confundirse con la *criptografía*, por ser ambas parte de los procesos de protección de información, son disciplinas distintas, tanto en su forma de implementar como en su objetivo mismo. La criptografía es la ciencia de la escritura enigmática, se utiliza para cifrar información de manera que sea ininteligible para un intruso subrepticio, pero éste mismo sabe que hay información cifrada, en cambio la esteganografía oculta la información en un portador de modo que no sea advertido el hecho mismo de su existencia y envío, y de esta forma, un intruso ni siquiera sabrá que se está transmitiendo información sensible. No obstante, la criptografía y la esteganografía podrían utilizarse de manera que una complemente a la otra.

El objetivo principal de la comunicación esteganográfica es ocultar y enviar información sensible a través de un portador, logrando que ésta pase totalmente inadvertida para quienes no participan de la comunicación encubierta.

Para que un *emisor* y un *receptor* logren una comunicación esteganográfica, es menester la existencia de diferentes *canales*: c*anal oculto o encubierto* y *canal público o legítimo*, donde el primero se define como el canal de comunicación en el que no se envía la información pertinente para la comunicación convencional. El segundo término o *canal público*, se refiere al ducto de datos que tienen como fin que la información de una comunicación convencional viaje por medio del mismo. Para ejemplificar, en una foto, una marca de agua no visible a simple vista en la foto pertenecería al canal oculto de comunicación, y la foto en sí al canal público.

El *emisor* es quien comienza el envío de mensajes por medio de un canal encubierto, siendo así, el *receptor* quien realizará el procedimiento inverso para obtener el mensaje.

Las comunicaciones encubiertas tienen sus bases dada la existencia de una interacción oculta entre el emisor y receptor. Esto implica básicamente, que existe una interacción a la vista de todos o pública, que explota una interacción oculta para comunicar un mensaje de manera disimulada a cierta porción de ese público. Por ende, a la vista de cualquiera simula que la comunicación real, consiste en los mensajes que actúan como *portadores*.

El *portador*, se define como el conjunto de datos que va a ser utilizado para adulterarlo. Al introducir un mensaje sensible, se mantiene oculto a partir de la aplicación del *algoritmo esteganográfico;* directamente relacionado con una *clave esteganográfica*, que permite determinar la forma en que se aplica. Tanto el emisor como el receptor, deberán predefinir antes de la comunicación el *algoritmo esteganográfico* con su *clave esteganográfica*.

Al acto de introducir un mensaje en un portador llamaremos *infiltrar*, mientras que al acto inverso, en donde se recupera la información oculta del portador se denominará *filtrar.*

El término *mensaje legítimo* se utilizará para referirse a la información transportada por el portador en sí, mientras que el mensaje a *infiltrar* en el acto esteganográfico se definirá como *mensaje sensible*.

Estos conceptos se entienden mejor a partir del relato de *Heródoto*, en su libro *Historias*, contextualizado en la rebelión de las ciudades jonias de Asia Menor, donde Aristágoras, tirano de Mileto, es solicitado por el rey de Persia a organizar una expedición contra la isla griega de Naxos. Pero Histaeus, griego, sabiendo que Aristágoras no permanecerá leal a Persia, quiere ponerse en contacto con él para pedirle que se una a la rebelión contra el rey persa. Para ello, hizo cortar el pelo al más leal de sus esclavos hasta dejarlo rapado, luego tatuó el mensaje secreto en su cabeza y esperó que le crezca el pelo. Con el pelo largo, el esclavo fue enviado a Aristágoras, pasando todos los controles, dado a que no llevaba nada extraño. Para leer el mensaje, Aristágoras le cortó el pelo nuevamente, culminando así con el éxito de la rebelión jonia ante el dominio persa.

Analizando el relato, el esclavo fue utilizado como *portador* por parte del *emisor*, *infiltrando* un *mensaje sensible* en su cabeza representada como *clave esteganográfica*, utilizando el acto de rapar la cabeza como *algoritmo esteganográfico* y al esclavo pasando los controles como el *mensaje legítimo*, *filtrado* posteriormente por el *receptor*.

Actualmente la esteganografía, desde una perspectiva moderna, y en términos informáticos, se refiere a la información o a un *archivo* cualquiera que se encuentre oculto dentro de otro, normalmente multimedial, es decir, el portador es una imagen, *archivo* de audio o de video. Se hace referencia a *archivo* de manera general para referirse a un conjunto de bits que conforman una entidad de información almacenada en algún lugar.

Haciendo referencia a la **Fig. 1**, el emisor comienza con el proceso de infiltración de un mensaje sensible dentro de un portador determinado, aplicando un algoritmo esteganográfico parametrizado con una clave esteganográfica, creando así un *mensaje esteganográfico*. Este último, es enviado al receptor, donde posteriormente a la recepción aplicará el algoritmo esteganográfico junto con la clave esteganográfica, llamando a esto el proceso de filtrado, obteniendo así el mensaje sensible.

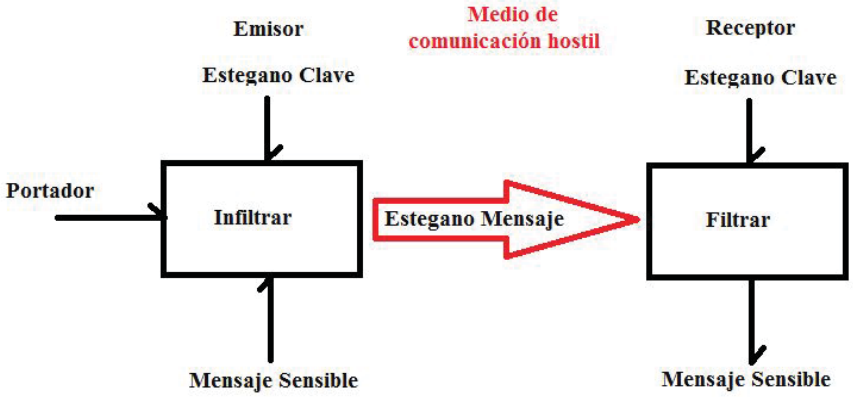

Fig. 1. Arquitectura de una comunicación esteganográfica.

Cualquier actor con acceso al medio de transmisión, tendrá acceso a los *mensajes esteganográficos*, pero solo podrá visualizar el *mensaje legítimo*, y no el *sensible*, ya que este último viaja abstractamente por el *canal oculto*, a diferencia del mensaje convencional que lo hace por medio del *canal legítimo.*

Un canal de comunicación permite explotar posibilidades de comunicaciones esteganográficas cuando es posible asignar otras interpretaciones a los campos que lo conforman, es decir, permite que en ciertos lugares fluya información diferente a la información que debería poseer por especificación. Por ejemplo, si la especificación técnica no aclara qué acción tomar si un campo determinado, toma el valor cero o uno en un contexto en particular.

En material legal, aún no se ha encontrado algún estándar que respalde o especifique a la esteganografía; lo que tiene sentido, ya que gran parte del éxito de una comunicación encubierta o esteganográfica depende estrictamente del factor "secreto", por lo que no tendría sentido adoptar o establecer un estándar esteganográfico. Dicho esto, los *portadores* de apariencia inocentes no despertaran sospechas a un *espía*, y los *mensajes sensibles* llegarán a destino sin problemas.

En la actualidad, los requerimientos y/o requisitos de un problema de comunicación segura, no toleran siquiera que un espía sospeche que se tiene algo que ocultar. No es suficiente con cifrar por métodos criptográficos la información que fluye en el medio, ya que con esto, lo único que se lograría es que el enemigo o espía no tenga acceso directo a dicha información, pero sabe de su existencia; lo que sería terriblemente problemático en situaciones de guerra, en donde, abundan secuestros, torturas, y abusos del poder.

Por otro lado, en el acto de una comunicación esteganográfica, donde el objetivo primordial es que el enemigo ignore la existencia de información; si éste sospecha de que la misma existe, o conoce el método esteganográfico utilizado, podrá extraerla sin demasiadas dificultades. No obstante, la idea principal es que a una primera vista, se presente la información en alguna forma aleatoria comprimiéndola y codificándola por algún método criptográfico previamente a la aplicación de alguna técnica esteganográfica. Sometiendo los *mensajes sensibles* a estos dos procesos, si un *espía esteganográfico* o criptoanalista sospecha de la existencia de los mismos, y conoce o logra descifrar el método esteganográfico utilizado, extraerá la información y se encontrará con un conjunto de bits de aspecto aleatorio, por lo tanto no probará con certeza si existe un mensaje sensible, si aplicó el método de *filtrado* correcto, o si simplemente extrajo "basura".

Queda a destacar, que dada la facilidad algorítmica de detección, y la complejidad de hallazgo que presenta la esteganografía, se caracterizará por ser sutilmente una gran *estrategia evasiva*. Se interpretará una *estrategia evasiva*, como el conjunto de acciones planificadas sistemáticamente en el tiempo, que se llevan a cabo para lograr un determinado fin o misión, para eludir cierta dificultad, compromiso o peligro.

De todas formas, no existe seguridad absoluta. Hoy en día, los espías experimentados son capaces de burlar lo impensado. Es por eso que dependerá de la habilidad o ingeniería personal que uno presente en el acto de ocultar información en una comunicación crítica o de la habilidad y astucia del espía que intenta descubrir la existencia de un canal encubierto de información.

3 End to End over IP

Desde la consolidación de internet las consideraciones iniciales que se tomaron en las definiciones del modelos OSI[1] se determinó que la función de los protocolos que coexisten sobre los niveles inferiores de un sistema de comunicación entre emisores y receptores intermedios deben encargarse del redireccionamiento y envío de los paquetes, sin preocuparse por su contenido, su seguridad y la correcta recepción.

El avance en las telecomunicaciones y el incremento de tráfico en las redes de datos, junto con una serie de hechos en torno a la seguridad hacen que el concepto descrito con anterioridad tenga que ser modificado o nuevamente evaluado dado que es factible sacar provecho de determinadas estructuras de protocolos dentro del stack TCP/IP[2] y así utilizar el mismo para fines a los que no fue creado.

Dentro estos protocolos de capas inferiores del modelo OSI se encuentra Internet Protocolo (IP) el cual vive en la capa de red y cuya responsabilidad radica en el envío y recepción de los paquetes de datos, dejando a capas superiores el control de errores y congestión. En otros términos, la red no posee inteligencia en cuanto a conexiones, retransmisiones, control de errores ni secuenciamiento y solo se encarga de transmitir lo que los extremos le indican, ya que cada uno de los paquetes de datos son tratados independientemente..

La capa de red provee el servicio de *direccionamiento*, que consiste básicamente en la transmisión de datagramas de bits llamados *paquetes de datos* entre dos extremos. Las *direcciones IP* poseen una longitud fija de treinta y dos bits, divididas en cuatro octetos. Están compuestas por una dirección de red y otra de host.

IP es un protocolo no orientado a la conexión lo que significa que no mantiene ningún registro de los datagramas sucesivos sino que cada uno es tratado de manera independiente , además comúnmente se lo denomina de mejor esfuerzo dado que todos los hosts dentro de la red reciben el mejor servicio posible en ese momento, lo que significa que obtendrán diferentes anchos de bandas y tiempos de respuesta en función del tráfico en la red. Dentro del encabezado IP que puede variar entre 20 y 40 bytes según se utilice o no el campo Opciones, tiene como único parámetro de seguridad una suma de control de cabecera que cumple el servicio de detectar errores que pueden haberse introducido durante la transmisión o el armado del mismo.

<table>
<tr><th>0-3</th><th>4-7</th><th>8-15</th><th>16-18</th><th>19-31</th></tr>
<tr><td>Versión</td><td>Tamaño Cabecera</td><td>Tipo de Servicio</td><td colspan="2">Longitud Total</td></tr>
<tr><td colspan="3">Identificador</td><td>Flags</td><td>Posición de Fragmento</td></tr>
<tr><td colspan="2">Time To Live</td><td>Protocolo</td><td colspan="2">Suma de Control de Cabecera</td></tr>
<tr><td colspan="5">Dirección IP de Origen</td></tr>
<tr><td colspan="5">Dirección IP de Destino</td></tr>
<tr><td colspan="4">Opciones</td><td>Relleno</td></tr>
</table>

Fig. 2. Cabecera Paquete de Datos IP.

4 Análisis de tráfico de red

En estos últimos años se han vivido muchos cambios desde el punto de vista de las comunicaciones digitales. No obstante, por el mismo desarrollo y masiva utilización, las redes son cada vez más susceptibles a ciberataques que mutan en función del tiempo.

Existen una gran variedad y diversidad de ciberataques posibles debido a la cantidad de protocolos y algoritmos que están en juego a la hora de enviar información y muchos de los cuales no fueron definidos centrándose en la seguridad. Por ello, se hace imprescindible diseñar un sistema de seguridad para tratar de detectar y mitigar posibles amenazas que pongan en riesgo la seguridad. Una de las principales herramientas que permite localizar intentos de ataques son los comúnmente llamados “sniffers”, que permiten explorar los

[1] Modelo de Interconexión de Sistemas Abiertos
[2] Protocolo de Control de Transmisión / Protocolo de Internet

datagramas dentro del stack TCP/IP. Los DPI[3] y DFI[4] son un claro ejemplo a la hora de hablar de analizadores de tráfico, utilizados hace varios años en aplicaciones especializadas en detección de spam, malware y ataques tipo DoS (Denial of Service), etc. En la literatura se suele hablar de DPI para hacer mención tanto al análisis como a las aplicaciones que hacen uso de ese estudio, a su vez previo a este concepto existe un tipo de análisis más simple denominado SPI[5]. Ambos difieren en que capa del modelo OSI se sitúan, haciendo esta aclaración con la **Fig. 3**.

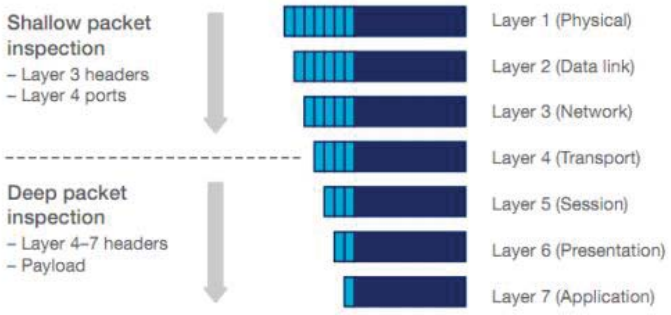

Fig. 3. DPI y SPI ubicados en las capas del modelo OSI

Por tanto, según esta definición se podría catalogar SPI como un conjunto reducido de DPI. Para evitar malentendidos, se limita el uso de SPI para el análisis hasta la cabecera de la capa de transporte, y el uso de DPI como el análisis más allá de la cabecera de ésta, es decir, el análisis de la capa de aplicación e incluso los datos de usuario.

La motivación para el desarrollo de DPI se fundamenta en que la clasificación tradicional basada en el análisis de las cabeceras de los niveles 2 a 4 del modelo OSI (SPI) no es un mecanismo fiable a día de hoy para determinar información sobre el protocolo y la aplicación que transporta un paquete.

DPI permite clasificar el tipo de tráfico en tiempo real, según el protocolo, pudiendo llegar a analizar en base a una base de firmas el tipo de información que se envía en el payload del datagrama.

El avance en la ingenieria de trafico trajo consigo nuevos conceptos como ser el enmascaramiento (técnica que utilizan ciertos protocolos que hacen un uso no acorde a los "puertos bien conocidos" para evitar su reconocimiento), y el cifrado (técnica por la cual se oculta información obteniendo un criptograma que solo puede ser legible por un tercero que conozca la clave). Debido a esto los métodos de análisis de tráfico nombrados se encontraron con deficiencias en el procesamiento de nuevos tipos de tráfico, por esta razón surge el desarrollo del análisis DFI , que lleva a cabo un análisis heurístico o de comportamiento para determinar qué tipo de tráfico está fluyendo por la red. DFI es capaz de detectar una aplicación (o amenaza) a partir del comportamiento del flujo de paquetes, en lugar de buscar el protocolo o el uso de puertos dentro del mismo paquete. Este hecho es importante, porque cada vez más tráfico está cifrado o es transmitido mediante 'tunneling' a través de la red, y más aplicaciones son enmascaradas.

Debido al caso de estudio de este paper se centrará en SPI, dado a que permite como hacer un estudio de los encabezados IP para extraer la información necesaria de los campos que lo componen.

5 Machine learning

El concepto de ciencia de datos engloba una serie de principios, problemas, definiciones y algoritmos para procesar y extraer información de un conjunto de datos (datasets) y encontrar en estos patrones.

La definición común entre las diferentes herramientas es la toma de decisiones en base a un análisis de datos que determina si los eventos se reiteran o estan relaciones en una serie de tiempo, pero también permite predecir comportamiento anómalo o extraño en función de un aprendizaje previo, y es en este punto en que el machine learning se convierte en herramienta para la investigación del siguiente estudio.

Para considerar la complejidad de la obtención de patrones se debe definir en relación con las habilidades humanas. Un humano puede definir reglas en base a pocas variables pero cuando ese número crece se vuelve

[3] Deep Packet Inspection
[4] Deep Flow Inspection
[5] Statefull Packet Inspection

inviable obtener una relación entre ellas. En consecuencia la ciencia de datos se aplica a la existencia de grandes volúmenes de variables y la clave del resultado es la correcta obtención de los atributos que se evaluarán.

EL machine learning (ML) se compone de dos fases. Primero se aplican algoritmos a los datasets para identificar patrones, estos se pueden representar de diferentes maneras como ser árboles de decisión, redes neuronales y modelos de regresión. Esta representación se la conoce como modelo. Luego, una vez que el modelo está creado se lo utiliza para el análisis. El desafío del ML es encontrar el algoritmo que mejor sesgo de aprendizaje obtenga para un conjunto de datos.

Dentro del ML se diferencian dos tipos de aprendizajes, supervisado y no supervisado. El aprendizaje supervisado trabaja con datos "etiquetados" intentando encontrar una función que dadas las variables de entrada (inputs) les asigne una etiqueta de salida adecuada. El algoritmo se entrena con un histórico de datos y así "aprende" a asignar la etiqueta de salida adecuada a un nuevo valor, es decir, predice el valor de salida. El otro tipo tiene lugar cuando no se dispone de datos "etiquetados" para el entrenamiento. Sólo conocemos los datos de entrada, pero no existen datos de salida que correspondan a un determinado input. Por tanto, sólo se podrá describir la estructura de los datos, para intentar encontrar algún tipo de organización que simplifique el análisis. por ello, tienen un carácter exploratorio.

Dentro de los algoritmos supervisados que componen el ML se utilizara árboles de decisión por su conveniencia ante problemas de clasificación. Estos se encuentran formados por nodos de decisión (está asociado a uno de los atributos y tiene 2 o más ramas que salen de él, cada una de ellas representando los posibles valores que puede tomar el atributo asociado) y nodos respuesta (está asociado a la clasificación que se quiere proporcionar, y devuelve la decisión del árbol con respecto al ejemplo de entrada).

En los arboles de decision, la decision para una instancia se toma comenzando por la parte superior del árbol y va navegando hacia abajo aplicando una secuencia de pruebas de atributos a la instancia. Cada nodo en el árbol especifica un atributo de prueba, y el proceso continúa nodo a nodo eligiendo qué rama seguir dependiendo del resultado de las pruebas de atributos en cada instancia.

El objetivo del algoritmo de aprendizaje del árbol de decisión es encontrar un conjunto de reglas de clasificación que divide el conjunto de datos de entrenamiento en conjuntos de instancias que tengan el mismo valor para el atributo objetivo.

6 Aplicación Esteganográfica en IPv4

Al analizar la estructura detallada del encabezado de este protocolo (explícito en la **Fig. 2**), el objetivo primordial será hallar segundas interpretaciones de los campos que lo conforman, es decir, interpretaciones que no estén contempladas en la especificación técnica formal del IP, sin que impacten de alguna forma "negativa" en la implementación del mismo. Esto servirá para saber que campos tener en cuenta a la hora de aplicar el algoritmo de aprendizaje.

Es de relevancia aclarar que la cabecera del *paquete de datos IP* resultante de una aplicación esteganográfica (con un *mensaje sensible oculto*) junto con su *carga útil*, será correctamente interpretado en todas las implementaciones de IP, es decir, respetará lo que indican los RFC.

Versión- Campo compuesto por 4 bits. Su objetivo es especificar la estructura que va a tener el paquete de datos, es decir, la estructura soportada por la versión especificada, que en este caso el número de versión será el 4 (cuatro), expresado en binario (*0100)*. Cada terminal o gateway al que llegue este paquete, verificará que el número de versión coincida con su estructura, por lo tanto, si se realiza alguna modificación, el campo no coincidirá con lo especificado, y el paquete IP será descartado. Entonces, este campo no se tendrá en cuenta.

Tamaño Cabecera- Este campo está también compuesto por cuatro bits, fue diseñado para especificar el largo en cuanto a cantidad de filas del encabezado del paquete de datos IP (cada fila con 32 bits). Según la especificación, el número mínimo de filas es "5", y éste, puede variar hasta como máximo "15" *(1111 en binario)* con el uso de un campo opcional denominado *Opciones*. Ya que el valor de este campo debe coincidir con el largo real del encabezado IP, entonces no permite aplicar técnicas esteganográficas alterando los valores, por lo tanto, no se tendrá en cuenta.

Identificador- Como bien se visualiza en la **Fig. 2**, es un campo de 16 bits, que si se lo eleva al cuadrado provee 65536 combinaciones posibles. Este campo sirve para asignar un identificador único a los paquetes de datos que serán fragmentados para viajar por la red debido a su gran tamaño. El protocolo IP, determina que todos los fragmentos que tengan el mismo identificador, *dirección IP de origen* y *dirección IP de destino* pertenecerán a un mismo paquete de datos, por lo tanto, de esta forma se logrará el reensamblado de los mismos. El procedimiento correcto para que este campo sea interpretado correctamente según su especificación técnica, simplemente consiste en que el emisor asigne un valor entre 0 y 65535 único en el momento que el paquete de

dato se encuentra "viajando" desde el emisor hasta el receptor. En concordancia, se puede inferir que en el *identificador* se podrá ingresar cualquier valor, siempre y cuando se cumpla con la condición de unicidad nombrada antes en ese contexto determinado. No obstante, es posible aplicar eficientemente esteganografía en este campo, ingresando en alguna combinación binaria que traducido a un lenguaje de alto nivel representa un carácter o símbolo.

Tipo de Servicio- Campo formado por 8 bits. El mismo expresa una métrica representada en 2^8 combinaciones posibles para determinar el tipo de servicio que se va a utilizar para la transmisión de los diferentes paquetes de datos en la red, en síntesis, asignar diferentes prioridades en el momento de envío.

En la especificación del protocolo, se puede apreciar que las diferentes combinaciones expresan diferentes opciones importantes de asignación de prioridades, pero lo relevante a destacar es que el bit 6 y el 7 fueron reservados para uso futuro. De acuerdo a ello, para esta versión, norma y contexto de utilización de protocolo, se podrá utilizar para esteganografiar por medio de sus 4 combinaciones (2^2).

Longitud Total- Este campo utiliza 16 bits para determinar el largo total del paquete de datos en sí, medidos en octetos, incluyendo encabezado y carga útil. La mayoría de las implementaciones IP son capaces de procesar paquetes de datos de hasta 576 octetos, por lo tanto, en la especificación se recomienda que los paquetes enviados no superen un tamaño de 64 octetos para la cabecera y 512 octetos para la carga útil (576 octetos en total). Al igual que el campo *Tamaño de Cabecera*, es un campo derivado, ya que va a depender del largo real del paquete de datos tratado en ese momento. No obstante, si llegase un paquete con un largo distinto al especificado en este campo, el gateway lo descartará.

Flags- Representa su funcionalidad mediante 3 bits, de los cuales el bit 1 (el bits que se encuentra en la posición 0, es decir, el más significativo) se encuentra reservado, y según la especificación, en dicho campo debe ir "0", pero no se aclara nada de qué sucede si dicho flag setea en "1". El segundo bit, si se encuentra en "1" significa que el paquete de datos no es fragmentable, y si está en "0", determina que el mismo se puede fragmentar. En general no se recomienda fragmentar dado a la alta demanda de procesamiento y recursos que requiere esta funcionalidad; pero si se da el caso de que este bit sea "1", y el router en ese momento decide que es necesario fragmentar (por condiciones del medio), pues entonces lo descartará. El bit 3 va a determinar la existencia o no de más fragmentos, por lo tanto, tiene sentido sólo si el bit "no fragmetar" se encuentra en "0". Aclarado esto, si el bit de "más fragmentos" es igual a "1", quiere decir que existen más fragmentos, y si está en "0", es el último fragmento, o que es lo mismo, no existen más fragmentos. Tampoco, en la documentación del protocolo, se especifica sobre qué sucede si el bit de "no fragmentar" y "más fragmentos" están en "1" en el mismo paquete de datos, por lo tanto, si se fuerza al mismo a que suceda esto, no se espera que se altere la integridad del paquete de datos, ni interfiera en la comunicación correcta del protocolo. Se definirá a *Flags* como posible a esteganografiar, ya que atenta con la semántica definida en la especificación técnica del protocolo, pero no rompe con la integridad del datagrama. Vale aclarar que con la modificación de este campo, se traería inconvenientes en la funcionalidad *Path MTU Discover*y[6], pero como no es considerada esencial, entonces podría darse por alto.

Posición de Fragmento- Formado por 13 bits. Básicamente determina a qué posición del paquete de datos pertenece este fragmento en particular. El primer fragmento poseerá la posición en "0", ya que este campo posee la posición del inicio del contenido del fragmento respecto del paquete de datos. Este parámetro, está directamente relacionado con el campo *Flags*, y en la especificación no aparecen determinaciones con respecto a que se debe hacer si el campo "no fragmentar" es "1" y en este campo existe un valor diferente a "0". *Posición de Fragmento*, tiene significado únicamente si el paquete de datos procesado es un fragmento, es decir, "no fragmentar" es "0". Según la especificación, se indica únicamente que si el campo "no fragmentar" es "1", entonces tanto *Posición de Fragmento* como el campo "más fragmentos", deben ir en "0". Este campo entonces posee posibilidades esteganográficas.

Si bien los dos últimos campos descritos, a simple vista no presentan un gran aporte en materia esteganográfica, una vez desarrollados los campos *Opciones* y *Suma de Control de Cabecera*, se podrá presentar tangiblemente un ejemplo de aplicación esteganográfica en el proceso de fragmentación y reensamblado de paquetes de datos, ya que la utilización de estos campos conforman el conjunto de información de control necesaria para que este procedimiento pueda llevarse a cabo.

Time to Live- Campo compuesto por 8 bits. El objetivo primordial es darle un tiempo de vida (TTL) a los paquetes de datos, y esto se cuantifica en la cantidad de gateway IP por los que se le está permitido "pasar" o ser procesados. Si éste se encuentra en "0", en el primer gateway al que llega el paquete de datos será destruido/descartado. Según la especificación, el TTL es medido en unidades de tiempo de segundos, y cada host con implementación IP, por la cual es procesado el paquete de datos, deberá descontar en al menos una

[6] Por sus siglas en inglés, Unidad Máxima de Transmisión, es una operación de control en donde se envían paquetes de datos con tamaños igual al máximo permitido.

unidad a este campo, por más que el tiempo de procesamiento del mismo sea diferente a una unidad de tiempo. El valor máximo que puede tomar este campo es 2^8, por lo tanto, si un paquete de datos posee este valor, significará que el mismo no podrá atravesar más de "255" gateways. Sobre las bases descritas, se podrá aplicar esteganografía pero tanto, el emisor como el receptor deberán interactuar en un escenario controlado y conocido, como por ejemplo, dado un contexto en el que el emisor y el receptor están separados por dos gateways, si el primero filtra un mensaje en este campo, se sabrá y definirá previamente que el paquete de datos pasará por medio de dos gateway, decrementandose este campo en dos unidades de tiempo al llegar al receptor. En síntesis, teniendo en cuenta las variables, se podría llegar a establecer un canal encubierto con este campo.

Protocolo- Con ocho bits describe el protocolo utilizado en el paquete de datos. Al tratar de modificar este campo, se alteraría la integridad y semántica del paquete. No obstante, no se podrá aplicar esteganografía.

Suma de Control de Cabecera- Se comenzará definiéndolo como no aplicable, ya que cumple con el objetivo de proteger al encabezado del paquete de datos de posibles errores de procesamientos o transmisión. Básicamente, aplica un algoritmo donde toma el encabezado del paquete en palabras de 16 bits (con este mismo campo en "0"), luego se calcula la suma aritmética del complemento a "1" (se suma bit a bit, y al resultado se le agrega el valor del acarreo de la suma) de todas las palabras, y se calcula el complemento a "1" (el/los "0/s" se cambian por "1/s", y el/los "1/s" por "0/s") del resultado. Este procedimiento se repite cada vez que el paquete se es procesado por los enrutadores.

Dirección Destino y *Dirección Origen*- Cada campo posee un tamaño de 32 bits. El primero determina la dirección del receptor, y el segundo, del emisor. Por cuestiones evidentes, no se utilizará este campo tampoco.

Opciones- Campo de longitud variable, puede tomar un valor máximo de 40 bytes, y en algunos paquetes de datos IP la existencia de este campo es optativa. Puede presentarse de dos formas: La primera, un único octeto que determina el tipo de opción. La segunda, es un octeto con el tipo de opción, opcionalmente un octeto con el largo de la opción, y el contenido de la opción. A su vez, el octeto perteneciente al tipo de opción está compuesto por tres partes: 1 bit correspondiente al *flag de copia*, los 2 siguientes determinan la *clase de opción*, y los 5 últimos codifican el *número de opción*. Al mismo tiempo, el *flag de copia* indica si esa opción se copia en todos los fragmentos (si es que existe fragmentación), donde si el mismo es igual a "0" indica que no se copia, y si vale "1" indica lo contrario. La *clase de opción*, posee 4 combinaciones posibles por el hecho de que está compuesto por 2 bits, y las mismas representan: *opción de control*, *reservado para uso futuro*, *opción de depuración y mediciones*, y la última combinación también es *reservado para uso futuro*. La especificación no determina el orden en que deben aparecer las diferentes combinaciones dentro de *clase de opción*. Con estas bases descriptas, será posible utilizar este ordenamiento para infiltrar un mensaje esteganográfico.

Relleno- Este campo se utiliza para asegurar de que el encabezado tenga una longitud igual a 32 bits, o más bien, que sea múltiplo de 4 octetos. Este campo se presenta como uno diferente dentro de la descripción del mismo en la especificación técnica de IP, y su estructura es igual a la primer combinación entre *clase de opción*, y *número de opción* descriptos en el campo anterior. No obstante, la esteganografía se aplicará de la misma forma que el campo anterior.

7 Caso de Estudio

El diseño del componente esteganográfico, fue realizado con las características necesarias para lograr una comunicación encubierta entre dos o más participantes. Aplicado a la práctica, se infiltraran los mensajes pertenecientes a la comunicación encubierta en paquetes de datos IP creados específicamente para realizar esta comunicación (proceso de infiltración, desde el lado del emisor), y posterior a la recepción. Con respecto al protocolo a utilizar en capas superiores se definió TCP, pero podría ser indiferente ya que el posterior análisis de flujos de datos será sobre la capa de red. En segundo lugar se diseñó un componente de machine learning donde se lo entrenó con tráfico específicamente generado para su customización, donde el mismo está construido con el algoritmo Random Forest.

Se generó un tráfico libre de anomalías, donde el mismo se montó sobre un entorno de red que a través de un port-mirror del switch se envía al "netflow probe" el tráfico con el cual generamos las estadísticas. Este dataset se enviará a un colector de datos que es donde posteriormente se realizarán los procesos de ML. Cómo trabajamos en un enfoque de anomalías, lo que se persigue es que las características definan cuál es el comportamiento normal del tráfico y por tanto, lo que se desvíe de ahí se considera anómalo. Estos se definió dado que al tener múltiples variantes de infiltración de información se haría muy dificultoso el entrenamiento del algoritmo para tráfico no benigno.

El caso de prueba se realizó en un ambiente controlado (Red LAN) en donde IP no sufre variaciones considerables y con esto descartamos una posible fragmentación de los datagramas que puedan distorsionar los resultados. En la **Fig. 4** se aprecia un diagrama arquitectónico de la solución.

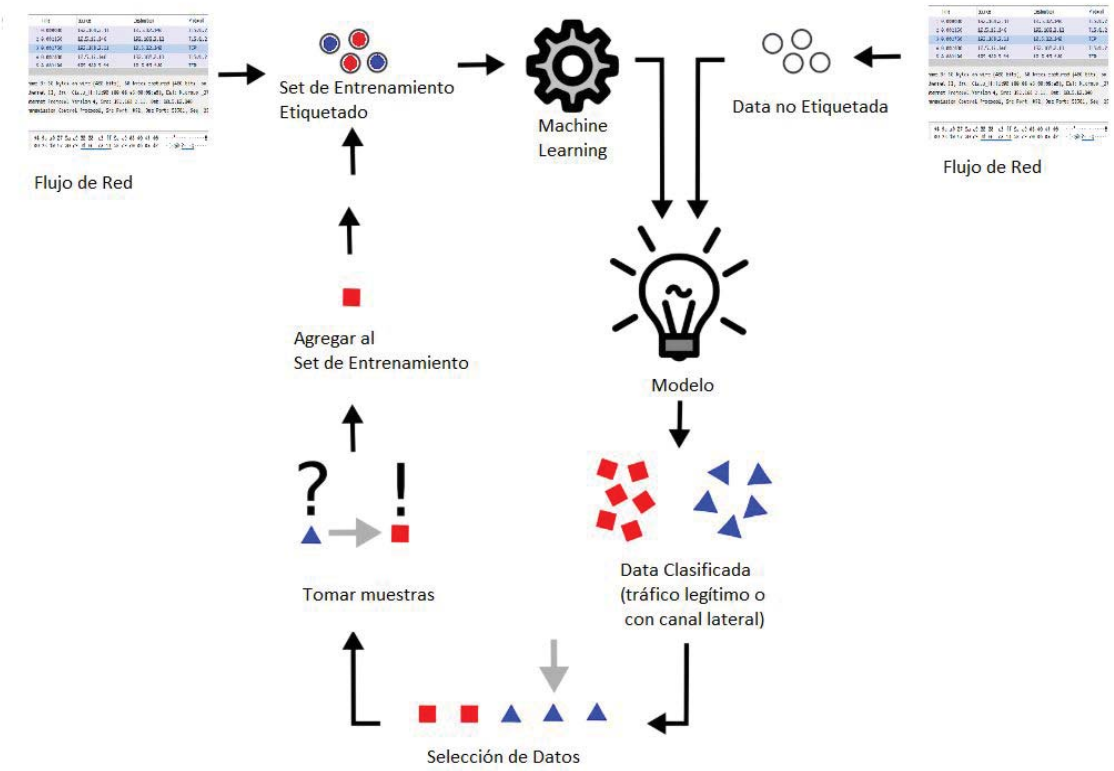

Fig. 4. Arquitectura para la detección de canales encubiertos.

8 Análisis de resultados

Partiendo de un volumen considerable de tráfico de red se entrenó el algoritmo random forest de aprendizaje en donde en cada nodo del árbol se evalúa un campo específico del header IP. En base a lo explicado en los apartados anteriores se determina si coincide con tráfico estipulado por el estándar del protocolo o existe una comunicación lateral.

Según los datos aprendidos por cada campo, se confeccionó en base a los aprendido un algoritmo de formato top-down en donde en cada instancia de nodos se compara a través de una estructura “if/else” si los datos fueron modificados intencionalmente o no. Dependiendo de la decisión tomada si es positiva la secuencia finaliza en una hoja o nodo de decisión, arrojando que el datagrama se vio comprometido, en caso contrario la iteración continua nodo a nodo evaluando los campos a analizar.

En las pruebas se obtuvo una efectividad de entre el 98% (con las tramas de 3 días de muestreo) y 91% (con las tramas de 5 días de muestreo) en la identificación de tráfico con canal lateral. Aún así, la versatilidad y los tiempos de ejecución más rápidos del algoritmo clasificador seleccionado lo convierten en una alternativa muy competitiva. Además, se estableció que el método propuesto en el caso de estudio y replicado sigue siendo útil en un contexto controlado dejando para futuras líneas de investigación diferentes topologías o entornos de red en donde entren en juego diferentes flujos de tráfico . El rendimiento de clasificación no puede competir con otros sistemas clasificadores, pero tiene la ventaja de ser totalmente no intrusivo, transparente y no requiere hardware adicional. El método seleccionado parece prometedor, y todavía hay espacio para mejoras adicionales en este contexto particular.

9 Conclusiones y Futuras líneas de investigación

En este trabajo de investigación, se estudió y analizó al Protocolo de Internet desde el punto de vista de su aplicabilidad esteganográfica, para así poder determinar e identificar los campos necesarios a analizar para detectar posibles comunicaciones encubiertas. En el desarrollo, se utilizan reglas o patrones no definidos por la especificación técnica del protocolo para realizar segundas interpretaciones que viola la política de seguridad del mismo. El objetivo principal fue detectar mensajes infiltrados en un canal de comunicación convencional por medio de la aplicación de algoritmos de machine learning.

Sobre estas bases estudiadas, se puede determinar que la posibilidad de aplicabilidad esteganográfica es un factor que se encuentra presente en el protocolo analizado y que también es totalmente detectable bajo la técnica implementada. Así mismo, estas características, deberían ser tenidas en cuenta por los encargados de gestionar la seguridad en el flujo de información en la red, ya que hoy en día no son muy conocidas las potencialidades de explotación de canales encubiertos, en caso de su utilización con fines no legales.

A partir de este estudio crítico de la especificación del Protocolo de Internet, implanta un punto de partida para posibles mejoras en el/los protocolo/s, y/o anticipar futuros inconvenientes en diferentes implementaciones.

Por último, como trabajos a futuro se definieron las siguientes:

a. Aplicación esteganográfica y detección de canales encubiertos en protocolos de otras capas, como por ejemplo: TCP (protocolo de capa de transporte), o HTTP (protocolo de capa de aplicación).
b. Implementar la detección de canales encubiertos por medio de otros algoritmos de machine learning y aprendizaje automático.
c. Aplicabilidad esteganográfica y detección de canales encubiertos en IPv6.
d. Abordar el estudio del diseño de un componente de análisis de flujo integrado donde también se analice la carga útil del protocolo de comunicaciones.
e. Investigar aplicabilidad para detección de tráfico malicioso y/o detección de malwares “0 days”.

Referencias

1. Tuomas A.: *“Invisibilidad Práctica en Comunicaciones Digitales”*. HUT Seminar on Network Security, (1995)
2. Cachin: *“Un Modelo de Información Teórico para Esteganografía”*, (2004)
3. Pitas I., Voyatzis G.: *“Chaotic mixing of digital images and applications to watermarking”*. European Conference on Multimedia Applications Services and Techniques, vol. 2, pp. 687–695, (1996)
4. RFC 791 Internet Protocol. https://tools.ietf.org/html/rfc791 . Accedido el 1 de Julio 2019.
5. Ahsan K. : *“Covert Channel Analysis and Data Hiding in TCP/IP”*, M.A.Sc. thesis, Dept. of Electrical and Computer Engineering, University of Toronto, (2002).
6. Lenti J.: *“STEGANOGRAPHIC METHODS”*, (2005)
7. Cole E.: “Ocultamiento de Vista Llana – Esteganografía y el Arte de la Comunicación Encubierta”, (2003)
8. Handel T., Sandford M.: *“Hiding data in the OSI network model”,* (1996)
9. Kalker T.: *“Defending Against Statistical Steganalysis”*, Niels Provos, (2002)
10. Cox J., Miller L., Bloom J. A., Fridrich J., Kalker T. : *“Marcas de Agua Digital y Esteganografía”*, (2008)
11. Shiroshita T., Takahashi O., Yamashita M., *“Integrating layered security into reliable multicast,* (1996)
12. Kipper G.: “Guía del Investigador de Esteganografía”, (2004)
13. Rowland C. H., *“Covert channels in the TCP/IP protocol suite”*, (1997)
14. Newman R.: *“Informática Oculta y Comunicaciones de la red”*, (2007)
15. Fabien A.P., Ross J.: *“En los Límites de la Esteganografía”*, (1997)
16. Stalling W.: “Comunicaciones y Redes de Computadores”. Décima Edición, (1997)
17. Blitz A.: *“Computer Forensisc and Investigations”*, Cuarta Edición, (2013)
18. Clair B.: “Esteganografía: Cómo enviar mensajes secretos”, (2001)