Universidad Nacional Autónoma de México

Posgrado en Ciencia e Ingeniería de la Computación

"Sistema de Detección de Intrusos Basado en Anomalías de Red Usando la Plataforma Numenta para Cómputo Inteligente"

T E S I S

que para obtener el grado de:

Maestro en Ciencias (Computación)

P R E S E N T A:

José Roberto Sánchez Soledad

Director de tesis:
Dr. Enrique Daltabuit Godas

México, D.F. 2010.

A mi esposa Maribel

Te dedico este trabajo de Tesis, siempre he contado con tu apoyo incondicional a lo largo de nuestro matrimonio, a tu lado encontré el Amor, la confianza, el apoyo y todos los sentimientos que han llenado mi corazón de alegría, gracias por ser la compañera de mi vida.

A mi hijo Roberto Alexis

Por traer la luz a mi vida, eres el mejor hijo que un padre podría desear. Doy gracias por cada día que te tengo a mi lado, te prometo que junto a tu madre nos esforzaremos por brindarte los valores y la educación que nuestros padres procuraron darnos.

Agradecimientos:

En mi camino he enfrentado a muchas dificultades que no se pudieron haber resueltos con solo mi necedad. Reconozco que debo un especial agradecimiento a quienes me han brindado un mundo de ayuda y, de vez en cuando, un mundo de inspiración.

A mis padres, Roberto y Virginia - Por brindarme la educación que me ha convertido en lo que soy, les agradezco por confiar en mí en todo momento, y alentarme a salir a delante.

A mis hermanos, Jesús y Guadalupe - Por creer en mi a lo largo de mi vida.

A mi Director de Tesis, Dr. Enrique Daltabuit - Primero por tener la paciencia de dirigir y revisar este trabajo de tesis, segundo por ser un ejemplo a seguir en todo momento, sus conocimientos me ayudaron a lograr esta tesis.

Resumen

Se presenta una descripción de los riesgos que existen en las redes de cómputo, así como los posibles ataques a los que se está expuesto.

Ofrece una descripción de los Sistemas de Detección de Intrusos que existen, así como sus componentes e implementaciones, haciendo énfasis en las ventajas de un IDS basado en anomalías o basado en Firmas.

Después de analizar las ventajas y desventajas de los métodos de detección que existen, se propone el análisis y desarrollo de un Sistema de Detección Basado en Anomalías de Red. Implicando directamente el uso de herramientas de inteligencia artificial que permitan el aprendizaje del comportamiento de una red de datos.

La tecnología de Memoria Temporal Jerárquica o HTM (Hierarchical Temporal Memory) es un nuevo paradigma de la inteligencia artificial basado en la estructura y función de la neocorteza del ser Humano. Las redes HTM son la piedra angular de las Plataforma para Cómputo Inteligente de Numenta (NuPIC), una nueva herramienta para el desarrollo de aplicaciones de inteligencia artificial. La creación de una red HTM para el análisis de tráfico implica modelar comportamiento de una red de computadoras.

Utilizando datos proporcionados por DARPA en la evaluación de sistemas de detección de intrusos, fue posible realizar el entrenamiento y pruebas del sistema de detección de intrusos utilizando NuPIC. Posteriormente utilizando una red de datos en producción, es decir, una red la cual está siendo utilizada, se implementó el IDS.

El desarrollo de un Sistema de Detección de Intrusos utilizando NuPIC pretende fijar las bases que permitan modelar el comportamiento de redes de cómputo, permitiendo la detección de amenazas en las anomalías de la red.

Abstract

A description of the risks that exist in computer networks is presented, as well as the possible attacks one is exposed to.

It offers a description of existing Intrusion Detection Systems (IDS), as well as their components and implementations, with an emphasis on the advantages of an anomaly- or firm-based IDS.

After an analysis of the pros and cons of the existing detection methods, an Anomaly-Based Detection System is proposed. It directly implies the use of Artificial Intelligence (AI) tools that allow learning of behavior observed within a data network.

The Hierarchical Temporal Memory Technology (HTM) is a new paradigm of AI based on the structure and function of a human brain's neocortex. HTM networks are the cornerstone for Numenta's Platform for Intelligent Computing (NuPIC), a new tool in the development of AI applications. The creation of an HTM network for traffic analysis implies a modeling of the behavior observed in a computer network.

Using data obtained from DARPA in the evaluation of IDS, it was possible to conduct training and testing of the IDS using NuPIC. Next, using a production-level network (meaning one that is actually being used), the IDS was implemented.

The development of an IDS using NuPIC seeks to set the foundations that allow to model the behavior of a data network, in turn enabling detection of threats from network anomalies.

Índice general

Índice de tablas

Índice de figuras

Prefacio

Dentro de los principales problemas que abarca la seguridad en cómputo el detectar eficazmente ataques de red sobre un sistema o una red misma se ha convertido en una problemática en la cual ya se encuentran disponibles distintos tipos de soluciones, en estas soluciones siguen existiendo falsos positivos, es decir alertas que no son tráfico malicioso pero son detectados como tal. Uno de los mecanismos que se utiliza para la detección de tráfico malicioso es la detección de anomalías, cuando existe un comportamiento normal del tráfico de la red e inesperadamente surge nuevo patrón se puede afirmar que existe una anomalía en el comportamiento de la red.

A lo largo del desarrollo de esta tesis se aborda el diseño y desarrollo de un Sistema de Detección de Intrusos (IDS) basado en anomalías.

El desarrollo de este IDS tiene la característica de detectar anomalías en el tráfico de una red de datos. Este trabajo propone una nueva implementación ya que realiza la detección de anomalías utilizando la plataforma Numenta para Cómputo Inteligente NuPIC, esta plataforma permite implementar redes Jerárquicas Temporales las cuales se basan en un nuevo paradigma de la Inteligencia Artificial.

Se decidió escoger la realización de la detección de anomalías utilizando HTM ya que permite que no se encontró ninguna implementación de un IDS basado en anomalías utilizando HTM, el trabajo expuesto a continuación, fue probado con datos de prueba proporcionados por DARPA, y por datos en una red en operación. La mayor contribución de este trabajo es el presentar un trabajo que forme las bases de la detección de anomalías de red, con el menor índice de falsos positivos en la detección.

La implementación de este sistema de detección de intrusos, permite detectar anomalías de red, las limitantes que se encontraron fue el procesamiento y la falta de memoria, ya que al tratarse de un sistema de aprendizaje, es necesario realizar una etapa de entrenamiento previa a la detección, el sistema es capaz de realizar una mejor predicción mientras más información de entrenamiento reciba, ya que a través de las redes de datos circula gran cantidad de información en el tiempo, es imposible alimentarla con todo el tráfico de red, por lo que se tomaron muestreos para el proceso de entrenamiento.

Este sistema de detección de anomalías trabaja de forma adecuada, pero es necesario continuar con el desarrollo de interfaces de usuario y otras etapas de entrenamiento, para conseguir un producto terminal, como se mencionó anteriormente este trabajo trata de sentar las bases de la detección de

anomalías de red con HTM, por lo que si se desea conseguir una herramienta de distribución libre o comercial es necesario continuar con la investigación y el desarrollo.

Capítulo 1

Introducción

1.1. Introducción

Al menos hasta finales de 1988 muy poca gente tomaba en serio el tema de la seguridad en redes de computadoras de propósito general. Mientras que por una parte Internet iba creciendo exponencialmente con redes importantes que se adherían a ella, como Bitnet[1] o Hepnet[2]; por otra parte, el auge de la informática de consumo (hasta la década de los ochenta muy poca gente contaba con una computadora y un módem en casa) unido a factores menos técnicos iba produciendo un aumento espectacular en el número de piratas informáticos [30].

El primer incidente de seguridad conocido fue el 22 de noviembre de 1988, el cual fue protagonizado por el famoso Internet Worm o gusano de Internet creado por Robert T. Morris. Miles de equipos de cómputo conectados a la red se vieron inutilizados durante días, cuyas perdidas se estimaron en millones de dólares. Desde ese momento el tema de la seguridad en sistemas operativos y redes ha sido un factor a tener muy en cuenta por cualquier responsable o administrador de sistemas informáticos [30]. Poco después de este incidente fue creado el CERT (Computer Emergency Response Team) en Carnegie Mellon, luego otras universidades y entidades organizaron sus propios equipos de respuesta a incidentes y solo después el gobierno de Estados Unidos adoptó el concepto a nivel nacional. El CERT es un grupo formado en su mayor parte por voluntarios cualificados de la comunidad informática, cuyo objetivo principal es facilitar una respuesta rápida a los problemas de seguridad que afecten a tanto a los equipos como a las redes de Internet.

Desde la creación del primer CERT hasta nuestros días se hace presente la preocupación por los temas relacionados a la seguridad en la red y a sus equipos. Los piratas de antaño casi han desaparecido, dando paso a nuevas generaciones de intrusos que forman grupos como Chaos Computer Club o Legion of Doom, organizan encuentros como Iberhack y editan revistas electrónicas (2600: The Hacker's Quartely o Phrack son quizás las más co-

[1]Bitnet era una antigua red internacional de computadoras de centros docentes y de investigación que ofrecía servicios interactivos de correo electrónico y de transferencia de archivos

[2]Red de física de alta energía. Red de investigación que se origina en los EE.UU. Pero que se ha extendido a la mayoría de los países, que involucra trabajos de física de alta energía. Entre los sitios más famosos se encuentran el Argonne National Laboratory, Brookhaven National Laboratory, Lawrence Berkeley Laboratory y el Stanford Linear Accelerator Center (SLAC).

nocidas, pero no las únicas). Todo esto con un objetivo principal: compartir conocimientos [30]. Si hace unos años cualquiera que quisiera adentrarse en el mundo de los hackers casi no tenía más remedio que conectar a alguna BBS [3] donde se tratara el tema, generalmente con una cantidad de información muy limitada, hoy en día tiene a su disposición gigabytes de información electrónica publicada en Internet; cualquier aprendiz de hacker puede conectarse a un servidor web, descargar un par de programas y ejecutarlos contra un servidor desprotegido y con un poco de suerte, esa misma persona puede conseguir un control total sobre un servidor, probablemente desde su equipo de escritorio y con una básica comprensión de lo que se esté realizando. Aunque sin grandes conocimientos técnicos, tienen a su disposición multitud de programas y documentos sobre seguridad (algo que los hackers de los ochenta apenas podían imaginar), además de equipos de cómputo potentes y conexiones a Internet baratas [30]. Por si esto fuera poco, se ven estimulados a través de sistemas de conversación como el IRC (Internet Relay Chat), donde en canales como #hack o #hackers presumen de sus logros ante sus colegas, o mediante páginas como Zone-H donde muestran el trabajo realizado durante sus intrusiones.

A la vista de lo comentado, parece claro que la seguridad de los equipos ha de ser algo a considerar en cualquier red. Diariamente circulan por la red todo tipo de datos, entre ellos muchos que se podrían catalogar como confidenciales (nóminas, expedientes, presupuestos, entre otros) o al menos como privados (correo electrónico, proyectos de investigación, artículos a punto de ser publicados, etc.).

1.2. Concepto de Seguridad

Podemos entender como seguridad una característica de cualquier sistema (informático o no) que nos indica que ese sistema está libre de todo peligro, daño o riesgo [26]. Para la mayoría de los expertos el concepto de seguridad en la informática es utópico porque no existe un sistema 100 % seguro. Para que

[3] Un Bulletin Board System o BBS (Sistema de Tablón de Anuncios) es un software para redes de computadoras que permite a los usuarios conectarse al sistema (a través de internet o a través de una línea telefónica) y utilizando un programa terminal (o telnet si es a través de internet), realizar funciones tales como descargar software y datos, leer noticias, intercambiar mensajes con otros usuarios, disfrutar de juegos en línea, leer los boletines, etc.

un sistema se pueda definir como seguro, debemos de dotar de las siguientes características al mismo [26]:

- ***C**onfidencialidad*
- ***I**ntegridad*
- ***D**isponibilidad*
- ***A**utenticidad*
- ***N**o repudio*

¿Qué implican cada uno de los aspectos de los que hablamos?

La confidencialidad nos dice que los objetos de un sistema han de ser accedidos únicamente por elementos autorizados a ello, y que esos elementos autorizados no van a convertir esa información en disponible para otras entidades [26]. Esta característica indica que la información únicamente va a ser accedida por las personas responsables y cualquier tipo acceso no autorizado ha de ser considerado una violación a la seguridad.

La integridad significa que los objetos sólo pueden ser modificados por elementos autorizados, y de una manera controlada [26].

La disponibilidad indica que los objetos del sistema tienen que permanecer accesibles a elementos autorizados [30].

La autenticidad es la propiedad que asegura el origen de la información. La identidad del emisor puede ser validada, de modo que se puede demostrar que es quien dice ser [26].

El no repudio es la propiedad que asegura que cualquier entidad que envía o recibe información no puede alegar ante terceros que no la envió o no la recibió [26].

1.3. Distinción entre amenaza y ataque

Una amenaza es un posible ataque en potencia ya que todo el tiempo se encuentra la posibilidad de atacar, la mayoría de las amenazas no las podemos controlar ya que dependen de factores externos o internos [30]. Podemos considerar amenazas internas a todas aquellas que se encuentran dentro de nuestro entorno de trabajo por ejemplo las personas que utilizan

directamente el software de la organización y amenazas externas a todas las que afectan directamente al sistema.

Podemos considerar un ataque a todo aquel evento, exitoso o no, que atenta sobre el buen funcionamiento del sistema de manera intencional [30], es decir cuando una amenaza es llevada a la acción, en la mayoría de las veces con un fin especifico.

1.4. Clasificación de los ataques

Entendemos por datos al conjunto de información lógica que manejan el software y el hardware, como por ejemplo paquetes que circulan por un cable de red o entradas de una base de datos [30]. Habitualmente los datos constituyen el principal elemento a proteger, ya que es el más amenazado y seguramente el más difícil de recuperar. Se pueden realizar multitud de ataques o dicho de otra forma, están expuestos a diferentes amenazas. Generalmente la taxonomía más elemental de estas amenazas las divide en cuatro grandes grupos: interrupción, intercepción, modificación y fabricación [26].

Un ataque se clasifica como interrupción si hace que un objeto del sistema se pierda, quede inutilizable o no disponible. Se tratará de una intercepción si un elemento no autorizado consigue un acceso a un determinado objeto del sistema y de una modificación si además de conseguir el acceso consigue modificar el objeto; algunos autores consideran un caso especial de la modificación: la destrucción, entendiéndola como una modificación que inutiliza al objeto afectado. Por último se dice que un ataque es una fabricación si se trata de una modificación destinada a conseguir un objeto similar al atacado de forma que sea difícil distinguir entre el objeto original y el 'fabricado'. En la figura 1.1 se muestran estos tipos de ataque de una forma gráfica, estas imágenes fueron obtenidas del libro [26].

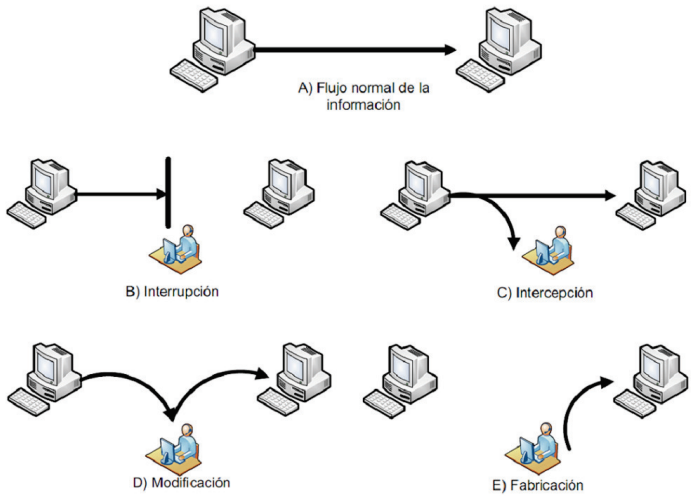

Figura 1.1: Tipos de Ataques de Red

Para poder protegerse de los distintos tipos de ataques que podrían poner en riesgo la seguridad de los sistemas existen diversos mecanismos para poder identificarlos y tomar acciones que ayuden mitigarlos, uno de ellos y del cual se origina la presente tesis son los mecanismos de detección, los cuales se utilizan para detectar violaciones de la seguridad o intentos de violación; ejemplos de estos mecanismos son los programas de auditoría, así como los sistemas de detección de intrusos (IDS).

El desarrollo de un Sistema de Detección de Intrusos permite detectar ataques de modificación y fabricación de datos que se observan en la figura 1.1.

1.5. Principales problemas de seguridad

Se le preguntó a la empresa Cibercorp, empresa mexicana especializada en establecer Soluciones de Tecnología y Consultoría ¿Cuáles son las principales problemáticas en materia de seguridad en México?. La empresa respondió que una problemática generalizada en materia de seguridad informática de nuestro país es la detección de ataques de red.

Teniendo en cuenta que la detección de ataques en la red es uno de los principales problemas en seguridad en nuestro país se elaboraron los siguientes objetivos.

1.6. Objetivos

Desarrollar una herramienta capaz de detectar ataques en una red de datos, esta herramienta debe generar alertas ante nuevas amenazas.

Como las nuevas amenazas son difíciles de detectar ya que en muchas ocasiones no existe registro de nuevos códigos o tráfico malicioso, la herramienta debe detectar anomalías de red, es decir, patrones de comportamiento anómalos los cuales pueden ser clasificados como posibles amenazas.

La herramienta debe ser capaz de generar pocos falsos positivos.

Podría ser colocada en modo pasivo para no generar latencia en el tráfico de red y tanto el tiempo de análisis como de procesamiento debe procurar ser mínimo.

Es necesario utilizar alguna técnica de inteligencia artificial que permita entrenar el IDS con el comportamiento normal del tráfico de una Red, como se explica en el capítulo 3, se decidió utilizar redes HTM, utilizando la Plataforma para Cómputo Inteligente de Numenta, no esta dentro de los objetivos de esta tesis profundizar en los algoritmos que utiliza la Plataforma.

1.6.1. Retos y Contribuciones

Los objetivos planteados describen una herramienta con distintas características, es importante mencionar que las características de la herramienta pretende desarrollar un IDS de distribución libre o comercial, el cual cumpla las necesidades de los usuarios. El alcance de este trabajo de tesis pretende establecer las bases de la detección de anomalías creando una red jerárquica mediante NuPIC. La utilización de una nueva tecnología como las redes jerárquicas temporales conlleva el poco desarrollo que existe en materia de implementación para los Sistemas de Detección de Intrusos.

La realización de este trabajo pretende contribuir a la investigación en el desarrollo de Sistema de Detección de Intrusos, además fija las bases de trabajos de tesis futuros en el campo de la detección de intrusos utilizando técnicas de inteligencia artificial.

El desarrollo de redes jerárquicas temporales que permitan el reconocimiento de patrones que ayuden a la identificación de tráfico malicioso y por ende la identificación de ataques sobre equipos o redes de información.

1.7. Estructura de la tesis

En el capítulo *Sistemas de Detección de Intrusos y Descripción de Protocolos de Red* se describen los sistemas de detección de intrusos así como los distintos tipos de sistemas que existen actualmente y la metodología que estos utilizan para la detección de intrusos, además se integra una descripción de los protocolos de red.(Capítulo. 2)

En el capítulo *Memoria Temporal Jerárquica* se presentan la teoría básica de como funciona este tipo de tecnología, así como conceptos y definiciones que se irán manejando a lo largo del desarrollo de esta Tesis.(Capítulo. 3)

En el capítulo *Implementación de HTM* se realiza la descripción del desarrollo del programa de tesis, así como la declaración de la red temporal jerárquica que se creó, además se describen los datos de prueba con los cuales se entrenó la red, y las pruebas que se realizaron tanto en un ambiente de pruebas, como en una red de computadoras reales.(Capítulo. 4)

En el capítulo *Pruebas* se muestran los resultados de las pruebas que se implementaron para la verificación del modelo de HTM propuesto en el capítulo 4.

En el capítulo *conclusiones* se dan a conocer las conclusiones de este trabajo de tesis, además se ofrece una perspectiva del trabajo futuro que ayudará al desarrollo de una herramienta más robusta que permita la distribución de la misma de forma libre o comercial.(Capítulo. 6)

Capítulo 2

IDS y Protocolos de Red

2.1. Estado del Arte

La idea de la detección de ataques surge desde 1980 en [1], trabajo en el cual se propone la detección de anomalías como un procedimiento valido, ya que los modelos de detección conocen lo que es 'normal' en la red o equipos de cómputo a lo largo del tiempo, desarrollando y actualizando patrones de comportamiento para compararlos con los eventos que se producen en los sistemas.

La creación del primer sistema experto en tiempo real utilizado para la detección de intrusos fue en 1991 con el desarrollo del sistema SRI IDES [12], desde entonces se han desarrollado sistemas que permiten observar el comportamiento en sistemas de computadoras y adaptar el aprendizaje del comportamiento normal para usuarios individuales, grupos y otros sistemas. Esta herramienta fijaba las bases para la detección de anomalías basadas en el comportamiento, ya que marca como una posible intrusión si el comportamiento se aparta significativamente de la conducta definida o incumple una regla en el sistema experto.

La creación de herramientas como NetSTAT [29] sistema de detección de intrusos orientado a redes permitió establecer los retos que conlleva el diseñar un sistema capaz de detectar intrusiones en ambientes como las redes de cómputo, esta herramienta plantea que diferentes eventos relacionados con una intrusión pueden ser visibles en diferentes lugares de la red. Además de mostrar los diferentes retos durante la identificación de tráfico malicioso como el variado tráfico que circula a través de la red.

Hay muchas técnicas que pueden ser aprovechadas para detectar el comportamiento anormal de tráfico en la red como se describe en [2]. Algunas de las cuales están basadas en examinar el contenido de los paquetes mientras que otras están basadas en estudiar la evolución del tráfico a través del tiempo. Muchos administradores de red han detectado anomalías interpretando el tráfico mediante gráficas utilizando herramientas de monitoreo como MRTG, CoralReef o ntop. El desarrollo de herramientas como SMARTCxAC permite la detección de anomalías usando predicciones en la adaptabilidad del tráfico junto con otras técnicas descritas en [2].

El proyecto Haystack [23] del Centro de Soporte Criptográfico de la Fuerza Aérea de Estados Unidos fue usado para ayudar a los oficiales a encontrar signos de ataques internos en los principales equipos de cómputo dentro de sus bases. El sistema fue escrito en ANSI C y SQL. Examinaba los datos de forma periódica almacenando colas de eventos para ser analizadas. Utilizaba

varias fases de análisis para detectar las posibles anomalías. El principal responsable del proyecto fue Steve Smaha.

El Laboratorio Nacional de Los Alamos en cooperación con el Laboratorio Nacional de Oak Ridge desarrollo un detector de anomalías llamado "Wisdom and Sense". Utilizaba técnicas no paramétricas ("nonparametric techniques"), son técnicas estadísticas que no hacían suposiciones sobre la distribución de los datos. Usaban este método para crear su propio conjunto de reglas. Luego analizaba las bitácoras de las auditorias en busca de excepciones de esas reglas, las cuales estaban organizadas en arreglos de datos con forma de árbol. Definían lo que era el comportamiento normal desde un punto de vista cronológico de los datos de auditoría [28].

Existen implementaciones de sistemas detección de intrusos basados en firmas, es decir únicamente identifican patrones de tráfico mediante bases de datos o reglas ya establecidas como SNORT uno de los más famosos por su gran efectividad además de ser de distribución libre, existen implementaciones comerciales también basadas en formas, como NFR (Network Flight Recorder) o ISS RealSecure [28], pero no son tan populares debido a su alto costo en el mercado.

Debido a la gran variedad de técnicas y desarrollo de Sistemas de Detección de Intrusos que fueron surgiendo DARPA realizó en 1998 una evaluación de los sistemas de detección de intrusos con la finalidad de satisfacer las necesidades de los investigadores, desarrolladores y en última instancia, los administradores de sistemas. Con lo cual desarrolló la primera evaluación real del rendimiento de sistemas de detección de intrusiones. El tráfico de red en la base de la Fuerza Aérea de Estados Unidos fue medido, caracterizado y posteriormente simulado en una red aislada en la que algunas computadoras se utilizaron para simular miles de diferentes sistemas Unix y cientos de usuarios diferentes en períodos de tráfico de red normal [9], estas muestras de tráfico se encuentran disponibles en la red.

Por último, cabe agregar que, buscando un mecanismo de detección de intrusiones distinto a los actuales, en este trabajo se presenta una propuesta que contempla el desarrollo e implementación de un Sistema de Detección de Intrusos basado en anomalías utilizando técnicas de inteligencia artificial como redes temporales jerárquicas.

2.2. Sistemas de Detección de Intrusos

El detectar intrusiones es el proceso de monitoreo de los eventos que ocurren en un sistema de computadoras o una red y analizarlos en busca de posibles incidentes [24], los cuales son violaciones o inminentes amenazas de las políticas de seguridad, políticas de uso aceptables, o practicas estándares de seguridad.

Los incidentes de seguridad ocurren debido a muchas causas como el malware (por ejemplo gusanos, spywers, etc.), los atacantes que intentan obtener acceso no autorizado a los sistemas de cómputo y los usuarios autorizados de los sistemas que hacen mal uso de privilegios.

Aunque muchos incidentes son de naturaleza maliciosa, muchos otros no lo son, por ejemplo una persona puede escribir de forma incorrecta una dirección ip a la que desea conectarse, con lo cual intenta acceder a un equipo al que no tiene acceso; esto es un claro ejemplo que no todos los intentos de conexión son maliciosos. Es indispensable contar con mecanismos que ayuden a la correcta detección de incidentes de seguridad.

Un Sistema de Detección de Intrusos (IDS) es software que automatiza el proceso de detección de intrusiones.

Existen varios tipos de IDS los cuales se diferencian principalmente por el tipo de eventos que pueden reconocer y las distintas metodologías que estos utilizan para detectar incidentes [24]. Los IDS comúnmente desempeñan las siguientes funciones:

- Registrar la información de los eventos observados [24]. Esta información usualmente es grabada localmente, y puede también ser enviada a sistemas por separado tal como un servidor de bitácoras centralizado, etc.

- Notificar a los administradores de seguridad de eventos importantes observados. Esta notificación es conocida como una alerta[1], se produce a través de varios métodos como pueden ser: e-mails, bitácoras, mensajes en alguna interfaz administrativa, paquetes de SNMP[2], mensajes de

[1] Notificación de una amenaza

[2] El Protocolo Simple de Administración de Red o SNMP es un protocolo de la capa de aplicación que facilita el intercambio de información de administración entre dispositivos de red. Es parte de la familia de protocolos TCP/IP. SNMP permite a los administradores supervisar el funcionamiento de la red, buscar y resolver sus problemas, y planear su crecimiento.

syslog, y programas o scritps definidos por el usuario [24]. Una notificación típicamente incluye solo información básica referente al evento.

Es importante resaltar que los Sistemas de Detección de Intrusos tienen como misión primordial la detección correcta de tráfico malicioso y no toman acciones preventivas o correctivas, es decir, no bloquean el tráfico o cortan la comunicación.

2.3. Metodologías de Detección

Las metodologías de detección que utilizan los IDS se pueden clasificar como basada en firmas o basada en anomalías, las cuales se describen a continuación.

2.3.1. Detección Basada en Firmas

Una *Firma* como se menciona en el documento [24] es un patrón que corresponde con una amenaza conocida. La *detección basada en firmas* es el proceso de comparar las firmas contra los eventos observados para identificar incidentes, contemplan secuencias de códigos binarios característicos de algún programa ejecutable. A continuación se muestran ejemplos de firmas [24]:

- Una conexión con telnet usando el nombre de usuario *root*, la cual es una violación a las políticas de seguridad de una organización debido a que nadie debería conectarse con ese usuario en el sistema.
- Un correo electrónico cuyo asunto sea *Imágenes Gratis* y adjuntado un archivo llamado *freepics.exe*, el envió de archivos ejecutables con extensión *.exe* es una característica de correos spam que intentan propagar software malicioso.

La detección basada en firmas es muy efectiva para detectar amenazas bien conocidas [16], pero es muy ineficiente para la detección de amenazas nuevas y sin registro alguno, existen muchas amenazas que utilizan técnicas para evadir estos procesos de detección, además existen variantes de amenazas conocidas por ejemplo si un atacante modifica el código malicioso mencionado anteriormente, y en lugar de colocar el nombre del archivo *freepics.exe*

coloca *freepics2.exe* esta técnica identificaría el primer nombre de archivo pero al modificar el nombre ya no podría ser posible su detección.

La detección basada en firmas es un método de detección simple por que solo compara una actividad actual [16], por ejemplo un paquete de tráfico de red se compararía contra una lista de firmas usando operaciones de comparación de cadenas.

Las tecnologías de la detección basada en firmas tienen poco entendimiento de la red o de los protocolos que esta utiliza [2], no pueden relacionar una petición de red con su respectiva respuesta. También carecen de la capacidad de recordar las peticiones anteriores en el momento que son procesadas las peticiones actuales. Esta limitación impide que sean detectados ataques que requieran de varios eventos si ninguno de los eventos contiene una indicación clara de un ataque, dicho de otra forma, una ataque puede requerir de varias peticiones para llegar a ser exitoso, pero al ser analizados evento por evento este método no puede detectar un ataque.

2.3.2. Detección basada en Anomalías

La detección basada en anomalías es el proceso de comparar la actividad que es considerada normal contra los eventos que están siendo observados para determinar variaciones significativas [24]. Un IDS que utiliza la detección basada en anomalías tiene *perfiles* [24] que representan el comportamiento normal de usuarios, equipos, conexiones de red y aplicaciones; por ejemplo un perfil para una red puede mostrar que la actividad de peticiones a paginas WEB abarca cerca del 13 % del *ancho de banda*[3] de una red durante las horas laborales en un día.

El IDS basado en anomalías usa métodos para comparar la característica de actividad actual con umbrales relacionados en el perfil, tal como el ejemplo del párrafo anterior el IDS podría detectar cuando la actividad de paginas Web ocupa un mayor ancho de banda de lo esperado y con esto alertar a un administrador de la anomalía. Los perfiles pueden ser desarrollados para muchos atributos del comportamiento, tales como el número de correos electrónicos enviados por un usuario, el número de intentos de acceso fallidos para un equipo, y el nivel de uso del procesador para un equipo en un período determinado de tiempo.

[3]Es común denominar ancho de banda digital a la cantidad de datos que se pueden transmitir en una unidad de tiempo. Por ejemplo, una línea ADSL de 256 kbps puede teóricamente, enviar 256000 bits por segundo.

El principal beneficio de los métodos de detección basados en anomalías es la eficacia en la detección de amenazas desconocidas [24]. Sí un equipo se infecta con un nuevo tipo de malware podría consumir recursos de procesamiento del equipo además enviar un gran número de correos electrónicos, lo que originaría gran cantidad de conexiones de red, y realizar otro tipo de comportamiento que sería significativamente diferente a los perfiles establecidos.

Un perfil establecido es generado sobre un periodo de tiempo(comúnmente días o hasta semanas), algunas veces también es conocido como *periodo de entrenamiento* [24]. Los perfiles para la detección de anomalías puede ser estáticos o dinámicos.

Un perfil dinámico es aquel que va cambiando con respecto al tiempo es ajustado constantemente ante eventos que son observados [24]. Esto se debe a que los sistemas y las redes cambian a través del tiempo, además el comportamiento normal también pueden cambiar.

Un perfil estático es aquel que nunca cambia, es generado una única vez con lo que eventualmente será inexacto por lo que es necesario regenerarlo periódicamente [24]. Los perfiles dinámicos no tienen este problema, pero ellos son susceptibles a intentos de evasión por parte de atacantes. Por ejemplo, un atacante puede realizar poca actividad maliciosa de vez en cuando posteriormente aumenta la frecuencia y la cantidad de actividad. Si la tasa de cambio es suficientemente lenta el IDS basado en anomalías puede pensar que la actividad maliciosa es un comportamiento normal, con lo que se incluirá en perfil dinámico.

Incluir actividad maliciosa dentro de un perfil es un problema común de los IDS basados en anomalías. En algunos casos los administradores pueden modificar el perfil para excluir la actividad que se sabe es maliciosa.

Otro problema con la construcción de los perfiles es que puede ser muy difícil en algunos casos, ya que la actividad de la red puede ser muy compleja. Por ejemplo, realizar un respaldo de información es posible que se realice una vez al mes y dicha actividad no sea colocada en el perfil. Al detectar tal actividad en la red el IDS levantará una alerta ya que no se encuentra en el perfil.

El mayor problema con los IDS basados en anomalías es que suelen producir *falsos positivos*[4] [2] debido a actividades no maliciosas que se apartan

[4]Los falsos positivos son alertas que se generan al detectar un tráfico valido, esto se debe a errores en la detección

significativamente de los perfiles establecidos, especialmente en entornos tan diversos y dinámicos como las Universidades.

2.4. Componentes y Arquitecturas

En esta sección se describe la mayoría de los componentes que acompañan a los IDS, y se ilustran las arquitecturas de red en las que son implementados.

2.4.1. Componentes típicos

A continuación se muestran los componentes típicos de un IDS descritos en el documento [24].

- *Agente o Sensor*. Los agentes o sensores de un IDS son los encargados del monitoreo y análisis del tráfico de red. Puede existir más de un agente en una implementación real, depende de la infraestructura de red.
- *Servidor de Administración*. Es un dispositivo que centraliza la información de sensores o agentes, y se encarga de la administración de los mismos.
- *Servidores de Base de Datos*. Es un repositorio donde es almacenada la información de los eventos que son recolectadas por los sensores.
- *Consola*. Es un programa que provee una interfaz a los usuarios y administradores. El software de consola es típicamente instalado en equipos estándar como laptops o equipos de escritorio.

2.4.2. Arquitecturas de Red

Los IDS pueden ser conectados entre sí, a través de la red de una organización o de una red separada que esta estrictamente diseñada para la gestión de software de seguridad conocida como una red de gestión [21], la separación de este tipo de redes se realiza mediante la segmentación lógica de la red a través de VLANs[5].

[5]Una VLAN (acrónimo de Virtual LAN, 'red de área local virtual') es un método de crear redes lógicamente independientes dentro de una misma red física

Es indispensable considerar el lugar donde se deben colocar los componentes del IDS, a continuación se muestran dos formas de colocar los sensores que se encargan de realizar el monitoreo y análisis del tráfico de red.

2.4.3. En línea

Un sensor en línea es colocado de forma tal que el tráfico de red pasa a través de él [24]. De hecho, algunos sensores en línea son híbridos de firewall e IDS, mientras que otros son simplemente IDS.

El principal motivo de colocar un sensor en línea es poder tomar acciones que mitigan el tráfico malicioso que es detectado. Los sensores en línea se colocan en los firewalls y otros dispositivos de seguridad, además es común colocar este tipo de sensores en las divisiones entre las redes, como las conexiones con redes externas y las fronteras entre las diferentes redes internas.

A continuación en la figura 2.1 se muestra el esquema de red ejemplificando un sensor en línea como se menciona en el documento [24].

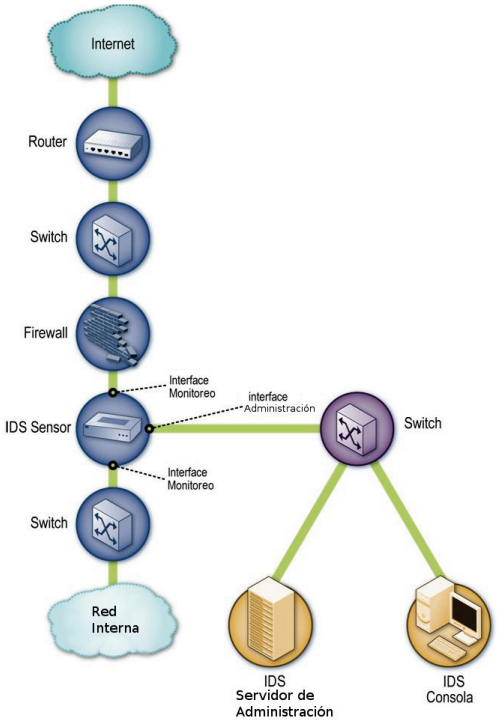

Figura 2.1: Arquitectura de Red de Sensor en Línea

2.4.4. Pasivo

Un sensor pasivo se implementa de forma tal que es posible analizar una copia del tráfico de red, no hay tráfico que pase a través del sensor [24]. Los sensores pasivos implementan un monitoreo en puntos clave de la red, tales como las divisiones entre las redes y segmentos de red críticos, como la actividad en una zona desmilitarizada (DMZ).

Los sensores pasivos utilizan otros dispositivos para poder observar una copia del tráfico de red por ejemplo el mostrado a continuación:

- *Port Mirror.* Muchos switch disponen de un puerto espejo [19], es decir un puerto que puede ver todo el tráfico de red pasa por él. La conexión de un sensor en un puerto espejo permite el monitoreo del tráfico hacia y desde muchos equipos. Utilizar este método de monitoreo es relativamente fácil y barato, ya que implica activar la funcionalidad que viene incluida en los switch.

A continuación en la figura 2.2 se muestra el esquema de red ejemplificando un sensor pasivo [24].

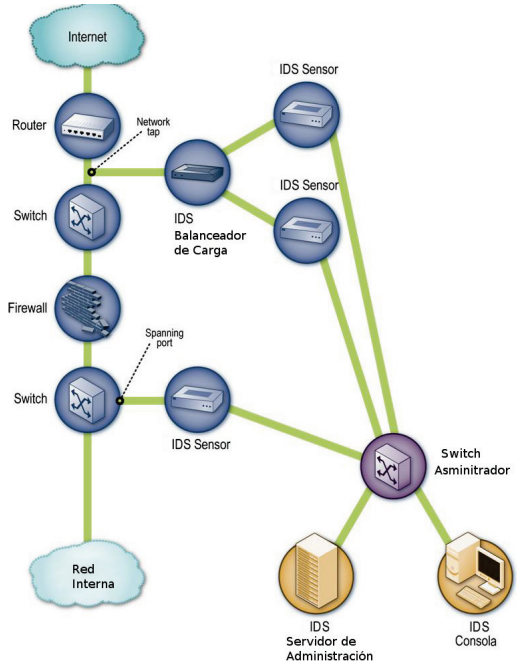

Figura 2.2: Arquitectura de Red de Sensor en Pasivo

La mayor debilidad que existe al utilizar un sensor pasivo es no permitir tomar acciones que mitigan las posibles amenazas en la red, es decir, al momento de detectar tráfico malicioso se realizará una acción de alerta, y posteriormente se bloquea ese tráfico a través de otros dispositivos como firewalls o listas de control de acceso.

En el capítulo 4 se mostrará el modelo propuesto para una implementación libre o comercial, lo cual no abarca el alcance de esta tesis.

2.5. Descripción de Protocolos de red

Existe gran variedad de paquetes que circulan por las redes de comunicaciones, cada paquete se encuentra descrito y conformado a través de un protocolo de comunicación, en las siguientes secciones se hace una descripción de los principales protocolos de comunicación que son empleados por las redes de computadoras. Esta descripción es necesaria ya que en el capítulo 4 se describen los datos se utilizarán para detectar tráfico malicioso.

2.6. Estándar IEEE 802.3 y Ethernet

El estándar IEEE 802.3 es para una LAN[6] CSMA/CD persistente-1. Este estándar tiene sus comienzos en el sistema ALOHA construido para permitir la comunicación por radio entre máquinas diseminadas por las islas hawaianas. Posteriormente se agregó detección de portadora, y Xerox PARC construyó un sistema CSMA/CD de 2.94 Mbps para conectar más de 100 estaciones de trabajo personales a un cable de 1 km [26].

La Ethernet de Xerox tuvo tanto éxito que Xerox, DEC e Intel diseñaron un estándar para una Ethernet de 10 Mbps. Este estándar formó la base del 802.3. El estándar publicado difiere de la especificación Ethernet en cuanto a que describe una familia completa de sistemas CSMA/CD persistente-1, operando a velocidades de 1 a 10 Mbps en varios medios. También, el único campo de cabecera difiere entre los dos (el campo de longitud del 802.3 se usa para el tipo de paquete en Ethernet [26].

Mucha gente usa el nombre de *Ethernet* en sentido genérico para referirse al 802.3, a lo largo de la tesis se utilizará indistintamente el nombre.

[6]Una red de área local, red local o LAN (del inglés Local Area Network) es la interconexión de varias computadoras y periféricos

A continuación en la figura 2.3 se muestra la descripción de la cabecera Ethernet.

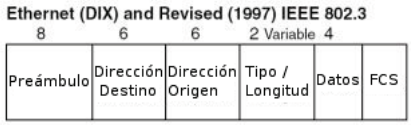

Figura 2.3: Cabecera Ethernet

Los campos que describen la cabecera Ethernet son los siguientes [26]:

- *Preámbulo*. Un campo de 7 bytes (56 bits) con una secuencia de bits usada para sincronizar y estabilizar el medio físico antes de iniciar la transmisión de datos.
- *SOF Inicio de Trama*. Patrón de 1s y 0s alternados que termina con dos 1s consecutivos.
- *Dirección de destino*. Campo de 6 bytes (48 bits) que especifica la dirección MAC hacia la que se envía la trama. Esta dirección de destino puede ser de un equipo.
- *Dirección de origen*. Campo de 6 bytes (48 bits) que especifica la dirección MAC desde la que se envía la trama.
- *Tipo*. Campo de 2 bytes (16 bits) que identifica el protocolo de red de alto nivel asociado con la trama o en su defecto, la longitud del campo de datos.
- *Datos*. Campo de 0 a 1500 bytes de longitud. Cada byte contiene una secuencia arbitraria de valores.
- *Relleno*. Campo de 0 a 46 bytes que se utiliza cuando la trama Ethernet no alcanza los 64 bytes mínimos para que no se presenten problemas de detección de colisiones cuando la trama es muy corta.
- *FCS (Frame Check Sequence - Secuencia de Verificación de Trama)*. Campo de 32 bits (4 bytes) que contiene un valor de verificación CRC[7].

[7]Control de redundancia cíclica

2.7. Protocolo IP

El protocolo IP o Internet Protocolo es el protocolo que define la forma en que los paquetes han de ser conformados para ser transmitidos por Internet.

Un datagrama IP consiste en una parte de cabecera y una parte de datos. La cabecera tiene una parte fija de 20 bytes y una parte opcional de longitud variable. En la figura 2.4 se muestra el formato de la cabecera IP, imagen tomada de [26].

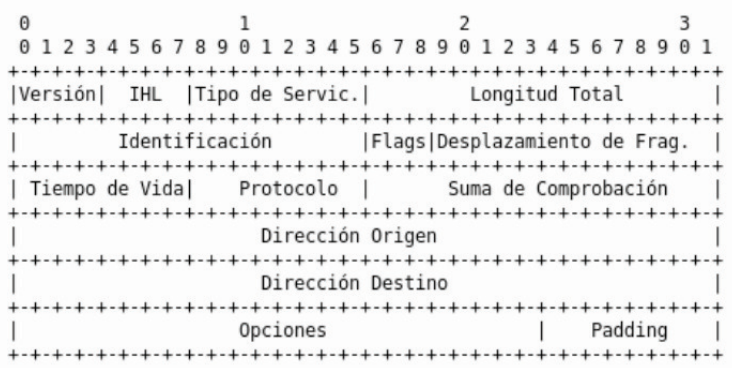

Figura 2.4: Cabecera IP

- *Versión*. Lleva la versión del registro al que pertenece el datagrama.
- *IHL*. Dado que la longitud de la cabecera no siempre es constante, se incluye este campo en la cabecera para indicar la longitud en palabras de 32 bits.
- *Tipo de servicio*. Permite al equipo indicar a la subred el tipo de servicio que quiere.
- *Longitud total*. Incluye todo el datagrama tanto la cabecera como los datos. La longitud máxima es de 65,535 bytes.
- *Identificación*. Es necesario para que el equipo destino determine a qué datagrama pertenece un fragmento recién llegado.
- *A continuación viene un bit sin uso.*

- *DF*. Significa no fragmentar, es una orden a los routeadores que no fragmenten el datagrama. Esto es común cuando el destino no es capaz de juntar las piezas de nuevo.
- *MF*. Significa más fragmentos. Todos los fragmentos menos el ultimo tienen este bit establecido.
- *Desplazamiento del fragmento*. Indica en que parte del datagrama actual va este fragmento.
- *Tiempo de vida*. Es un contador que sirve para limitar la vida de un paquete. Se supone que este contador cuenta el tiempo en segundos, permitiendo una vida máxima de 255 seg; debe disminuirse en cada salto y se supone que disminuye muchas veces al encolarse un tiempo grande en un router.
- *Protocolo*. Indica la capa de transporte a la que debe entregarse.
- *Suma de verificación*. Verifica solamente la cabecera.
- *Dirección de origen y destino*. Indica el número de red y el número de equipo.
- *Opciones*. Este campo se rellena para completar múltiplos de cuatro bytes.

Es importante hacer la descripción de este protocolo ya que ciertos campos se utilizarán para la detección de tráfico anómalo.

2.8. Protocolo TCP

El protocolo TCP(Transmission Control Protocol, Protocolo de control de Transmisión) se diseñó específicamente para proporcionar un flujo de bytes confiable a través de una red. Permite adaptarse dinámicamente a las propiedades de las distintas redes y es robusto ante muchos tipos de fallas [26]. Este protocolo es sumamente importante ya que paginas web, servicios de correo electrónico, resolución de nombres de dominio, y muchas otras aplicaciones utilizan este protocolo para comunicarse.

A continuación en la figura 2.5 se muestra las cabeceras TCP, imagen obtenida de [26].

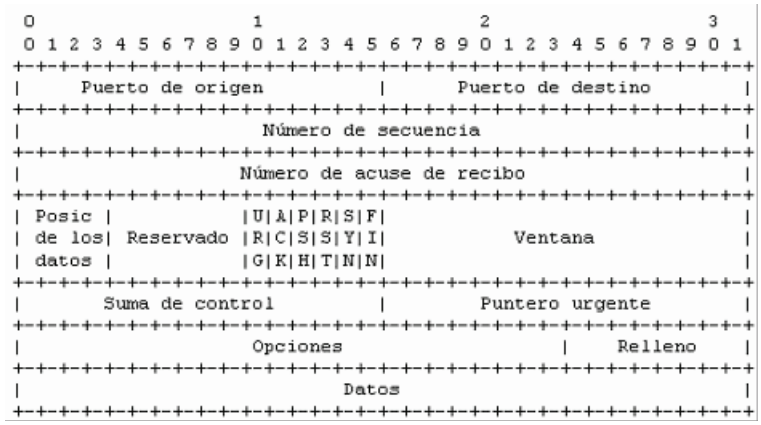

Figura 2.5: Cabecera TCP

- *Puerto Origen* y *Puerto Destino*. Identifican los puertos locales de la conexión.
- *Número de Secuencia* y *Número de Acuse de recibo*. Desempeñan sus funciones normales, es decir el orden en el que cada paquete es enviado y su número de recibo.
- *Longitud de cabecera TCP*. Indica la cantidad de palabras de 32 bits contenidas en la cabecera TCP. Esta información es necesaria porque el campo de opciones es de longitud variable, por lo que la cabecera también lo es.
- *A continuación se encuentra un campo de 6 bits sin utilizar*
- *URG*. Se establece en 1 si esta en uso el apuntador de urgente.
- *ACK*. Este bit se establece en 1 para indicar que el número de acuse de recibo es válido. Si el ACK es 0, el segmento no contiene un acuse de recibo, por lo que se ignora el campo de número de acuse de recibo.
- *PHS*. Este bit indica los datos empujados con push. Por este medio se solicita atentamente el receptor entregar los datos a la aplicación de su llegada y no ponerlos en buffer hasta la recepción de un buffer completo.
- *RST* . Este bit se usa para establecer una conexión que se ha confundido debido a la caída de un equipo u otra razón. Por lo general, si se recibe

un segmento con el bit RST encendido se tiene un problema en la comunicación.

- *SYN*. Se usa para establecer conexiones.
- *FIN*. Se usa para liberar una conexión. Especifica que el emisor no tiene más datos que transmitir.
- *Tamaño de la ventana*. El control de flujo en el protocolo TCP se maneja usando una ventana, es decir la cantidad de bytes que pueden enviarse.
- *Suma de comprobación*. Es una suma de comprobación de la cabecera, los datos y la pseudo cabecera conceptual mostrada en la figura anterior.
- *Opciones*. Este campo se diseño para contar con una manera de agregar características extras no cubiertas por la cabecera normal.

Capítulo 3

Memoria Temporal Jerárquica

3.1. On intelligence

Como Jeff Hawkins describe en su libro [7], hay muchas cosas que los humanos encuentran fáciles de hacer que las computadoras actualmente no pueden realizar tan sencillo. Tareas como el reconocimiento de patrones visuales, la comprensión del lenguaje hablado, el reconocimiento y manipulación de objetos a través del tacto.

Con el paso del tiempo se han realizado algoritmos para llevar a cabo las tareas antes mencionadas, desafortunadamente no se cuentan con algoritmos tan viables para realizar estas y otras funciones cognitivas[1] en un equipo de cómputo [7].

En un ser humano estas capacidades son en gran parte realizadas por la neo corteza cerebral [7], debido a este punto Jeff Hawkins realizó una teoría llamada Memoria - Predicción, la cual se encuentra fundamentada en el funcionamiento de la neo corteza cerebral.

Hawkins propone que "el cerebro oye, ve, entiende el lenguaje, e incluso juega ajedrez usando una sola herramienta" [7]. Hawkins ha ido más lejos y propuso que una jerarquía asociativa basada en patrones espaciales y temporales es responsable de todo el almacenamiento de memoria y comportamientos cognitivos en los seres humanos [4].

3.2. Memoria - Predicción

Jeff Hawkins es ingeniero informático inventor del Palm Pilot y del teléfono celular Treo, fundador de las empresas Palm y Handspring. Además ha trabajado en el campo de la neurociencia y es presidente del Instituto de Neurociencia de Redwood, fundado por él en 2002. Junto con Donna Dubinsky y Dileep George ha fundado la empresa Numenta, con el objetivo de desarrollar un nuevo tipo de memoria basada en el funcionamiento del cerebro humano.

Hawkins escribió el libro llamado *On Intelligence* [7] en el cual propone una teoría para el funcionamiento de la corteza que no se fundamenta exclusivamente en el campo de la biología, esta teoría indica que también tiene sentido desde un punto de vista computacional, la teoría es llamada Memoria-Predicción.

[1]Por cognitivo entendemos el acto de conocimiento, en sus acciones de almacenar, recuperar, reconocer, comprender, organizar y usar la información recibida a través de los sentidos.

A continuación se muestran los puntos principales de la teoría Memoria-Predicción y su relación con la neo corteza cerebral, los cuales pueden ser profundizados en [3]:

- La neo corteza construye un modelo para patrones espaciales y temporales a los cuales se está expuesto [3]. El objetivo de la construcción de los modelos es la predicción del siguiente patrón de entrada.
- La corteza se construye mediante la reproducción de una unidad de cómputo básica conocida como el circuito canónico de la corteza [3]. Desde un punto de vista computacional este circuito canónico puede ser tratado como un nodo que se repite varias veces.
- La corteza se organiza como una jerarquía [3]. Esto significa que los nodos[2] están conectados en un árbol en forma de jerarquía.
- La función de la corteza es modelar el mundo al que está expuesto [3]. Es decir obtener una representación de patrones.
- La neo corteza construye el modelo del mundo de manera no supervisada [3].
- Cada nodo en la jerarquía almacena una gran cantidad de patrones y secuencias [3]. El método de reconocimiento de patrones empleados por la corteza se basa principalmente en el almacenamiento de gran cantidad de patrones.
- La salida de un nodo se encuentra en términos de las secuencias de patrones que ha aprendido [3].
- La información es enviada a arriba y abajo en la jerarquía para reconocer y eliminar la ambigüedad de la información [3]. Este procedimiento es utilizado para predecir el siguiente patrón de entrada.

Para resumir esta teoría Hawkins plantea el siguiente ejemplo tomado de [4]. Los sensores del ser humano (ojos, oídos, piel, etc.) envían una señal a una neurona considerada como un elemento de la memoria (o nodo) en el cerebro. Si el sensor es un ojo y este ve un perro, una señal que representa "Los dientes del perro"va a ser enviada a un nodo en la memoria. Nodos

[2]Los nodos, según la teoría de Hawkins son las unidades básicas de cálculo

de memoria cercanos pueden enviar otras señales para la representación de los ojos, los labios, etc. Figura 3.1 Nivel 1. Los nodos envían lo que ellos están percibiendo hacia un nodo superior en la memoria, la cual percibe una mandíbula Figura 3.1 Nivel 2. El proceso continúa hasta puede es posible ver un perro Figura 3.1 Nivel 3.

En cada nivel los nodos superiores envían información de vuelta a los nodos inferiores, esencialmente diciendo "He visto esto antes, es una 'cabeza de perro' y si vez al final del perro puedes esperar ver una cola". Esta información es transmitida a la parte inferior de la jerarquía como sub representaciones.

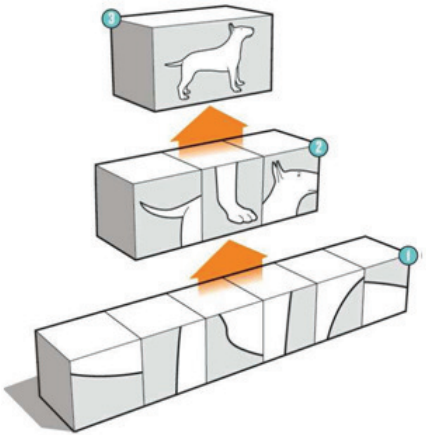

Figura 3.1: Ejemplo de la teoría Memoria-Predicción

Traslapando este ejemplo al análisis de tráfico, es decir, si en vez de tratar de identificar un perro se utilizan tramas de tráfico de red, es posible detectar tráfico malicioso.

Si se utiliza un sensor que observe los fragmentos de un paquete de tráfico que circula por la red, este sensor enviaría la información a un nodo superior de lo que el creería estar observando en esos momentos, el nodo superior utilizaría la información de los nodos inferiores para determinar el tipo de paquete que está observando y así poder pasar la información a un nodo superior que determine el tipo de tráfico figura 3.2.

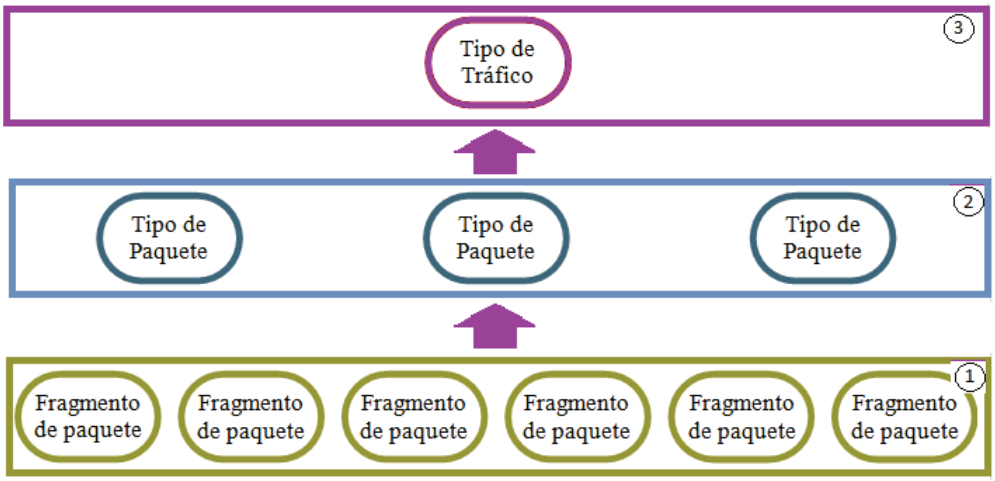

Figura 3.2: Ejemplo en el análisis de tráfico

En las siguientes secciones se describe el modelo de aprendizaje desarrollado por Hawkins el cual tiene por nombre "Memoria Temporal Jerárquica", el cual refleja su teoría Memoria-Predicción.

Memoria Temporal Jerárquica (HTM) es una tecnología que reproduce la propiedad estructural y algorítmica de la corteza cerebral. HTM ofrece la promesa de construir máquinas que se acercan o superan el rendimiento a nivel humano para muchas de las tareas cognitivas ya que permitiría almacenar, recuperar, reconocer, comprender, organizar y usar la información recibida a través de los sentidos [4].

Sin embargo hasta el día de hoy las redes HTM sólo se han utilizado para desarrollar pocos sistemas, incluyendo el reconocimiento de patrones de visión que se limitan a reconocer un pequeño conjunto de dibujos [4]. El objetivo de la investigación presentada en esta tesis es ofrecer una implementación de la tecnología HTM para identificar patrones de tráfico malicioso.

A continuación se muestran conceptos de las HTM y el cómo se utilizará para nuestro caso de estudio.

3.2.1. Descubrir las causas del mundo

Según [4] el *Mundo* consiste de objetos y su relación entre estos, algunos objetos en el mundo son físicos como carros, personas y edificios. Mientras que otros no son tangibles como ideas, palabras, sonidos, etc.

La teoría de HTM llama *Causas* a los objetos en el mundo.

Ya que el mundo es muy amplio una red HTM puede solo considerar un subconjunto de este mundo. Una HTM puede ser restringida al conocimiento financiero, al entendimiento de los datos geofísicos, demográficos, etc. [4]. En nuestro caso de estudio se restringe al reconocimiento de anomalías en una red de datos.

El mayor atributo del *Mundo* [4] desde el punto de vista de la teoría de HTM es que el mundo existe en el tiempo.

A lo largo de esta tesis se considera al mundo como una red de datos y las causas dentro de este mundo son los paquetes que circulan a través de está, las causas del mundo son representadas en forma vectorial, por lo que es necesario generar un vector con la información del tráfico de red.

Todos los sistemas HTM necesitan ir a través de una fase de aprendizaje donde el HTM aprende las causas del mundo. Todas las HTM primero aprenden acerca de pequeñas y simples causas de su mundo [4]. Durante la fase de entrenamiento la HTM aprenderá sus casusas, es decir el comportamiento normal de tráfico.

Con el suficiente entrenamiento y el diseño apropiado, es posible construir HTMs que descubran causas que los seres humanos no puedan descubrir [5]. Después del entrenamiento inicial una HTM puede continuar aprendiendo o no, dependiendo de las necesidades de la aplicación.

3.2.2. Creencias

Las HTM reciben patrones espaciales y temporales desde un sensor. En primera instancia las HTM no tienen conocimiento de las causas del mundo, pero a través del proceso de aprendizaje descubre que causas lo conforman [4]. El objetivo final de este proceso es el desarrollo de una representación de las causas en el mundo. Dentro del cerebro las células nerviosas aprenden la representación de las causas en el mundo. En una HTM las causas son representadas por números en un vector en cualquier momento en el tiempo, dada una entrada actual una HTM va a asignar una probabilidad individual de las causas que están siendo censadas [5]. Dicho de otra forma los nodos que conforman la HTM van a determinar las probabilidades de que una causa este siendo censada, en el ejemplo del perro la HTM determinará la probabilidad de ver una mandíbula o una cola y en nuestro caso de estudio la HTM determinará la probabilidad de estar viendo un paquete de tráfico normal.

La salida de las HTM es manifestada en un conjunto de probabilidades para cada una de las causas aprendidas. Este proceso es conocido como *creencia*.

3.2.3. Inferir causas de una nueva entrada

Después de realizar el aprendizaje de las causas en el mundo y como es que son representadas se puede realizar el proceso de inferencia.

La inferencia es similar al reconocimiento de patrones, dado una nueva entrada en el sensor una HTM, va a *Inferir* que causas conocidas son probables que se presenten en el mundo en ese momento [4]. Por ejemplo si se tiene un sistema de visión basado en HTM es posible mostrar figuras y las HTM infieren que objetos están en la figura. Si se tiene un sistema para analizar tráfico de red la HTM va a inferir el tipo de paquetes que la conforman.

En la mayoría de los sistemas basados en HTM la entrada del sensor siempre va a ser un nuevo dato. En nuestro caso de estudio el nuevo dato representa un paquete del tráfico en la red.

3.2.4. Realizar predicciones

Los sistemas HTM consisten de una jerarquía de nodos donde cada nodo aprende causas y creencias. Parte del algoritmo de aprendizaje para cada nodo es almacenar posibles secuencias de patrones [4].

Mediante la combinación de las posibles secuencias con entradas actuales cada nodo tiene la habilidad de realizar predicciones de que dato tiene mayor probabilidad de que ocurra [5]. Una HTM puede inferir causas de una nueva entrada. Esto permite también hacer predicciones acerca de eventos nuevos.

3.2.5. Aprendizaje supervisado y no supervisado

Cuando las redes HTM son entrenadas es posible agregar información de categoría al nodo y permitir su clasificación, por lo tanto el nodo puede realizar agrupaciones de información acorde a la categoría (Aprendizaje supervisado) [5].

Si no se incluye la información de categoría (Aprendizaje no supervisado), el clasificador crea grupos basado en las características de la entrada de los datos [5].

El desarrollo del sistema de detección de intrusos se realizó utilizando aprendizaje no supervisado, debido a la dificultad de llevar a cabo una clasificación de cada paquete de tráfico.

Al realizar el aprendizaje no supervisado la red HTM crea agrupaciones de tráfico figura 3.3.

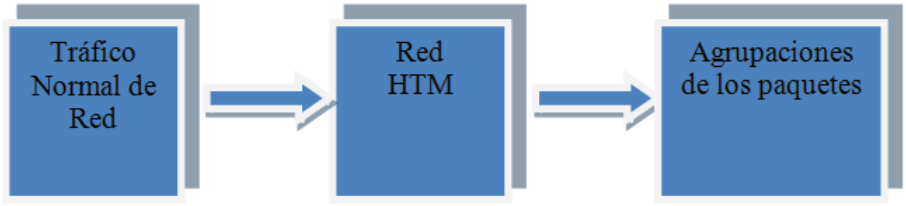

Figura 3.3: Agrupaciones de tráfico

Durante la fase de pruebas o proceso de inferencia se utilizan las agrupaciones creadas, y la red HTM tratará de agrupar los nuevos datos en estás, si no encuentran agrupaciones que encajen con su patrón la red HTM creará nuevas agrupaciones. El tráfico de red que se encuentre en estas nuevas agrupaciones se considera tráfico anómalo, ya que durante la fase de entrenamiento no se tenía registrado este patrón.

3.3. Nodos

Existen dos tipos de nodos en una red HTM, los nodos de memoria y los nodos sensores.

Nodos de Memoria. Son nodos que procesan los datos censados usando algoritmos espaciales y temporales [5]. Si un patrón es reconocido por un nodo la representación es enviada a la parte superior de la jerarquía, si el nodo observa un nuevo dato es decir no logra identificarlo entonces una representación es creada y enviada a la parte superior como una nueva representación.

Por ejemplo [4], si se desea construir una red HTM para regular y monitorar el tráfico de una autopista. Un enfoque podría ser la instalación de sensores a lo largo de las carreteras principales y sensores para monitorear la distancia entre los coches. Un concepto fundamental de HTM es que los datos de entrada deben contener componentes tanto espaciales como temporales,

un análisis de los datos de tráfico cumplen con esta norma. El aspecto temporal de los datos de tráfico está ilustrado por el "efecto domino", un conductor puede provocar el tráfico en muchos kilómetros con el simple hecho de frenar su vehículo. Este fenómeno sería casi imposible de detectar usando los datos recolectados en un instante de tiempo, y sólo es posible observarlo mediante cambios en los datos en distintos puntos en el tiempo. El componente espacial se puede encontrar mediante el análisis de las distancias entre los coches. A diferencia del "efecto domin", sería posible examinar datos en un instante de tiempo utilizando un pequeño número de sensores e identificar los coches que están estacionados.

Nodos sensor. Son nodos los cuales envían datos espaciales, basados en una entrada, arriba de la jerarquía de los nodos de memoria [5]. Esto es el equivalente a los ojos o oídos en los seres humanos.

El objetivo de cada nodo en una red HTM es formar un modelo para describir los datos que son encontrados. Cada nodo contiene un contenedor espacial y un contenedor temporal que trabajan juntos para identificar patrones en el tiempo [4].

La entrada del nodo está conectada al contenedor espacial. La salida del contenedor espacial se conecta a la entrada del contenedor temporal, que es el responsable de la salida del nodo. Estos mecanismos de contención se utilizan para construir un modelo de los datos encontrados, y permiten que el nodo realice inferencias sobre nuevos datos.

Un contenedor espacial se puede considerar como un proceso de cuantificación que asigna un número potencialmente infinito de patrones de entrada a un número finito de centros de cuantificación.

A continuación en la figura 3.4 se puede apreciar la estructura de un Nodo HTM.

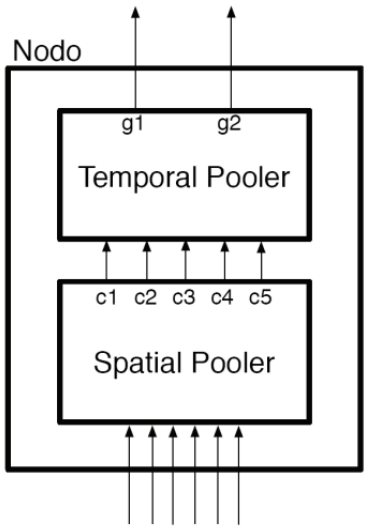

Figura 3.4: Estructura de Nodo HTM

Una vez que los datos entran en un nodo, es responsabilidad del contenedor espacial cuantificar las entradas de datos en un intento de eliminar el ruido y agrupar patrones de entrada de datos similares [18]

El contenedor espacial lleva a cabo esta tarea utilizando una función de distancia de Gauss para determinar la distancia entre vectores de n-longitud de datos [18]. Esta distancia se compara con el umbral de cuantificación y se toma una decisión en cuanto a cómo cuantificar el vector.

La sensibilidad del contenedor espacial puede ser ajustada utilizando los parámetros de configuración del nodo y puede variar dependiendo de su posición en la jerarquía.

En la Figura 3.5 se muestran vectores de datos, y la forma en que el contenedor espacial realiza la cuantificación de los datos, figura tomada de [18].

Distancia entre [1 1 1 1 1] y [1 1 0.99 1 1.02]

$$0.0224 = \sqrt{(1-1)^2 + (1-1)^2 + (1-0.99)^2 + (1-1)^2 + (1-1.02)^2}$$

Umbral de cuantificación = 0.1, Vectores iguales

Distancia entre [1 1 1 1 1] y [1 3 1.5 1 0.1]

$$2.2494 = \sqrt{(1-1)^2 + (1-3)^2 + (1-1.5)^2 + (1-1)^2 + (1-0.1)^2}$$

Umbral de cuantificación = 0.1, Vectores iguales

Figura 3.5: Cuantificación del contenedor Espacial

Los datos cuantificados son la entrada del contenedor temporal, el cual trata de agrupar patrones basados en su proximidad temporal [5]. Cuando un nodo está en modo de aprendizaje, registra que los patrones se encuentran cerca entre sí en tiempo y almacena los datos en una matriz de adyacencia de tiempos. Figura 3.6 muestra cómo la matriz está formada, y como almacena la información de proximidad temporal, imagen tomada de [5].

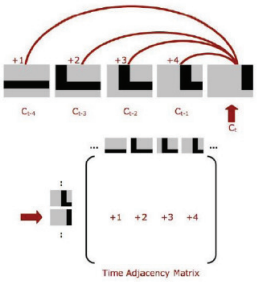

Figura 3.6: Matriz de Adyacencias

Cuando un nodo se cambia a modo de inferencia, se examina la matriz e intenta agrupar patrones con alta proximidad temporal. El contenedor temporal logra esto mediante la conversión de la matriz de adyacencia de tiempos en un grafo no dirigido [5]. Los pesos de las aristas dependen de su proximidad temporal. Los grupos temporales se identifican examinando los componentes conectados y sus agrupaciones basados en sus aristas. Estos grupos se utilizan cuando se intenta formar creencias sobre nuevos datos y la salida de un nodo esta en términos de los grupos temporales que ha aprendido [5].

En la Figura 3.7 se muestran varias iteraciones del proceso de agrupación temporal, imagen tomada de [5].

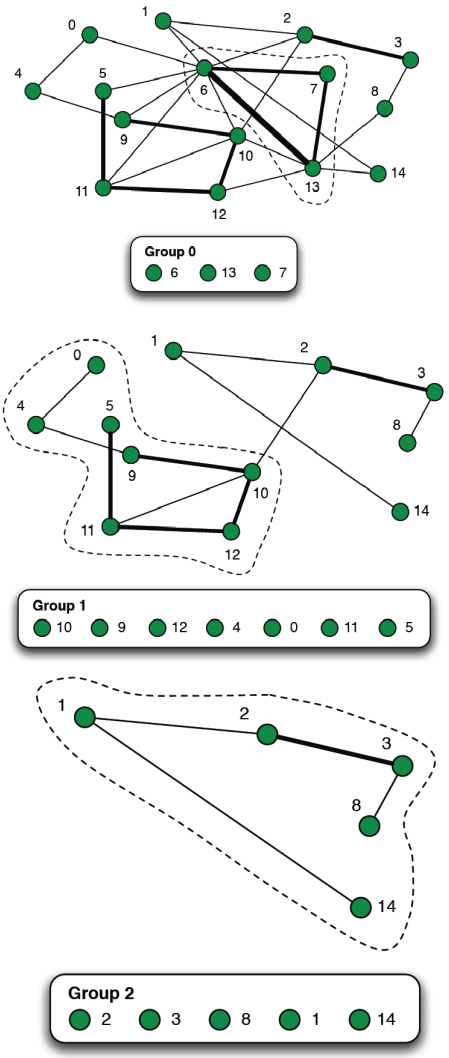

Figura 3.7: Agrupación Temporal

A continuación se muestra en la figura 3.8 un ejemplo sencillo, en el cual se tiene la representación de caracteres con forma de Montaña, Valle y Pradera, imágenes tomadas de [3]

Figura 3.8: Ejemplo sensor

En la figura 3.9 se muestra la red HTM capaz de reconocer entre las imagenes de la figura 3.8 con forma de Montaña, Valle y Pradera.

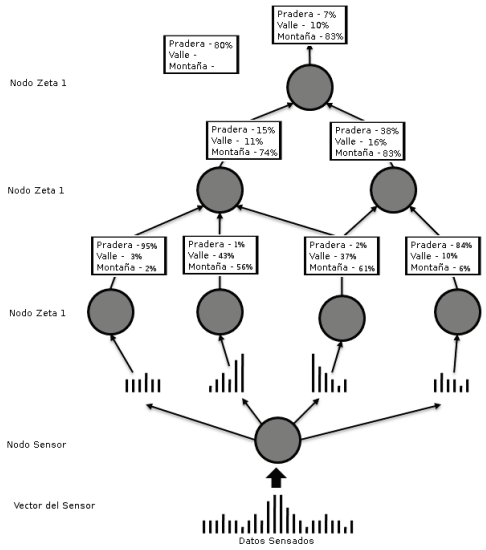

Figura 3.9: Ejemplo de red HTM

En el ejemplo anterior figura 3.9 se puede apreciar que la entrada de datos del sensor es una Montaña, posteriormente es pasado al sensor de datos el cual descompone el vector en cuatro partes que a su vez son enviadas a los nodos superiores, donde se obtiene la creencia de cada nodo la cual es pasada a los nodos superiores, hasta llegar al nodo superior donde determina que se trata de una montaña.

3.4. NuPIC

La tecnología HTM es un nuevo paradigma de cómputo basado en la estructura de la función de la neocorteza. Las redes HTM son la piedra angular de la tecnología aplicada en la Plataforma Numenta para Cómputo Inteligente, una nueva herramienta para la inteligencia artificial desarrollada por la empresa Numenta. Nupic provee de una serie de herramientas que se pueden usar para entrenar y prueba de redes HTM.

La implementación de la HTM para el desarrollo del detector de intrusos fue realizada mediante el framework NuPIC. El cual consiste de módulos de programación en python.

No se encuentra dentro de los alcances de esta tesis profundizar en los algoritmos utilizados por NuPIC, únicamente se utilizó el framework para desarrollar el IDS basado en anomalías.

Capítulo 4

Diseño e Implementación

4.1. Definición del Problema

Como se describió en la introducción los ataques a través de las redes de computadoras tienen un gran impacto en la vida de las personas hoy en día, por ejemplo si el servidor de un banco es comprometido el intruso podría quitarnos todo nuestro dinero, o si un virus informático logra infectar nuestra computadora podríamos llegar a perder toda la información de la misma, estas son algunas de las razones por las que detectar ataques y tomar acciones que los mitiguen se ha vuelto indispensable.

En el Capitulo 2 se describen los tipos de IDS que existen, así como la tecnología que utilizan para la detección de ataques además se mencionan las implicaciones de utilizar una u otra tecnología.

Para el desarrollo de este IDS se tomó la decisión de utilizar un método de detección basado en anomalías de red. Es en este punto donde entran las Redes Jerárquicas Temporales (HTM) utilizando esta tecnología que implementa NuPIC[1].

La creación de una red Jerarquica Temporal capaz de aprender tráfico normal y que permita la identificación de anomalías es sin lugar a duda el mayor reto del trabajo de tesis. Implica la modelación del mundo en terminos de patrones de red.

En primera instancia se utilizarán datos de prueba para el desarrollo, entrenamiento y pruebas del IDS.

Posteriormente se realizará la implementación del IDS en un ambiente real, donde se estudia su desempeño en la detección de anomalías, las cuales en el proceso de detección de intrusos son consideradas posibles ataques.

4.1.1. Implicaciones

Existen infinidad de datos viajando a través de una red de computadoras, estos datos utilizan protocolos[2] que permiten que las computadoras se entiendan, estos protocolos abarcan desde el tipo de acceso al medio como Ethernet, Token Ring, FDDI Fiber Distributed Data Interface o PPP Point-to-Point Protocol, por mencionar algunos, hasta la forma en como se envían

[1] Plataforma para Cómputo Inteligente Nupic

[2] Los protocolos de comunicación para la comunicación digital por redes de computadoras tienen características destinadas a asegurar un intercambio de datos fiable a través de un canal de comunicación imperfecto. Los protocolos de comunicación siguen ciertas reglas para que el sistema funcione apropiadamente.

los datos.

4.2. Hipótesis

En los capítulos anteriores se abordaron dos temas principales para esté trabajo de tesis, el diseño de sistemas de detección de intrusos y el estudio de las redes HTM. Con base en los capítulos anteriores se establece como hipótesis que la creación de una red HTM permitirá modelar el comportamiento de una red de cómputo, clasificando el tráfico en agrupaciones y permitiendo la detección de anomalías de red utilizando dichas agrupaciones. Durante el proceso de entrenamiento la red HTM creará agrupaciones del tráfico normal de la red, al llevar acabo el proceso de inferencia la red HTM clasificará el nuevo tráfico en las categorías previamente creadas, al encontrar nuevos patrones de comportamiento generará nuevas agrupaciones las cuales se considerarán anomalías del tráfico, dentro de dichas anomalías se encuentran los ataques a la red.

4.3. Datos de Prueba

El Grupo de Tecnología de Sistemas de Información (IST) del Laboratorio Lincoln del MIT, en colaboración con la Agencia de Proyectos de Investigación Avanzada de la Defensa de los Estados Unidos (DARPA)[3] y el Laboratorio de Investigación de la Fuerza Aérea (AFRL / SNS), ha recogido y distribuido los datos de tráfico de red para la evaluación de sistemas de detección de intrusos.

Los datos que se ponen a disposición del público a través de la pagina web `http://www.ll.mit.edu/mission/communications/ist/corpora/ideval/index.html`, son capturas de tráfico de red realizadas con el snifer[4] conocido como tcpdump.

[3]DARPA es acrónimo de la expresión en inglés Defense Advanced Research Projects Agency (Agencia de Proyectos de Investigación Avanzada de la Defensa) es una agencia del Departamento de Defensa de los Estados Unidos responsable del desarrollo de nuevas tecnologías para uso militar. Fue creada en 1958 como consecuencia tecnológica de la llamada Guerra Fría, y del que surgieron, una década después, los fundamentos de ARPANET, red que dio origen a Internet.

[4]Analizador de Protocolos

Las capturas de tráfico de red consisten de 5 semanas de capturas de tráfico, en las cuales se realizaron ataques en distintas semanas como se muestra a continuación.

- *Semana 1.* Durante esta semana se capturó el tráfico de entrada y de salida de la red, no se realizaron ataques a la red.
- *Semana 2.* Se realizó la misma captura del tráfico de la red, pero se realizaron ataques esporádicos a la red.
- *Semana 3.* Durante esta semana se capturó el tráfico de entrada y de salida de la red, no se realizaron ataques a la red.
- *Semana 4.* Se realizó la misma captura del tráfico de la red, pero se realizaron ataques esporádicos a la red.
- *Semana 5.* Se realizó la misma captura del tráfico de la red, pero se realizaron ataques esporádicos a la red.

El primer paso para el desarrollo del IDS es determinar el comportamiento normal de la red de datos, para el proceso de entrenamiento se utilizarán los datos de prueba de las semanas 1 y 3, que son considerados tráfico normal. Las semanas 2, 4 y 5 son consideradas capturas con tráfico anómalo, es decir, cuentan con ataques.

Considerando la gran cantidad de protocolos que circulan por la red es necesario hacer una clasificación de los mismos, se utilizarán únicamente los protocolos Ethernet, TCP e IP. Debido a que son los protocolos más utilizados en los equipos de cómputo.

Después de realizar el análisis de los datos de cada protocolo se muestra a continuación los datos necesarios por protocolo para llevar a acabo el reconocimiento de tráfico malicioso.

En el caso del protocolo Ethernet su descripción muestra que gran parte de los campos permanecerán constantes durante su transmisión, ya que el estándar así lo indica, además se puede observar que hay información que puede ser muy variada como lo es la información contenida en los datos, o el código de Verificación de Trama, es por esta razón que se déside tomar únicamente los datos de *Dirección origen*, *Dirección destino* y *tipo*, a los que en lo posterior se hará referencia como Direcciones MAC[5] y tipo.

[5]Media Access Control address

- ETHERNET
  - Dirección MAC Origen
  - Dirección MAC Destino
  - Tipo

  Así como en el protocolo Ethernet, en este protocolo existen datos muy variables y otros que permanecen constantes, es por esta misma razón que se déside tomar únicamente los datos de *Dirección IP origen*, *Dirección IP destino*, *Longitud total*, *identificación*, *protocolo*, y *Banderas IP*, las direcciones ip origen y destino ayudarán a determinar los equipos que se encuentran generando tráfico malicioso, además de dar el comportamiento normal de las conexiones en la red, es decir, permiten formar el perfil de conexiones entre equipos.

- IP
  - Dirección IP Origen
  - Dirección IP Destino
  - Longitud total
  - Identificación
  - Protocolo
  - Banderas

  De igual forma que en los protocolos anteriores en el protocolo TCP no son necesarios todos los campos, por lo que se seleccionaron los siguientes campos *Puerto Origen*, *Puerto Destino*, *Número de Secuencia*, *Número de Acuse de Recibo ACK*,*Banderas TCP*, y *Tamaño de la ventana*. Esta información permitirá saber el origen de los programas que generan el tráfico malicioso, por ejemplo si el destino de un ataque es el puerto 22, podría tratarse de un equipo que se intenta acceder remotamente a un equipo.

- TCP
  - Puerto Origen
  - Puerto Destino

- Numero de Secuencia
- Número ACK
- Banderas TCP
- Ventana

4.4. Desarrollo de Vector del Sensor

En el capítulo anterior se describe la definición de un sensor y aplicación de un sensor en NUPIC. Por lo que a continuación se muestra desarrollo del sensor para la detección de tráfico.

Se realizó el diseño de la red HTM para procesar los datos de DARPA, como datos de entrenamiento y posteriormente como datos de prueba y así llevar acabo el proceso de inferencia, es importante aclarar que para cada red de datos que se desee analizar es indispensable realizar el entrenamiento con los datos de la red a analizar, ya que es necesario desarrollar un perfil de datos de cada red, si se desea probar el IDS en una empresa es necesario realizar un perfil con un muestreo del tráfico que se considera como normal, para poder llevar acabo el entrenamiento de la red.

Como los datos proporcionados por la evaluación de IDS de la colección de DARPA, fueron recolectados mediante el sniffer tcpdump, se decidió utilizar esta misma herramienta par poder leerlos, como se muestra a continuación.

En el siguiente comando se puede apreciar que se utilizaron las opciones:

- *-nn* Evita la resolución de nombres con direcciones IP.
- *-ttt* Imprime una Delta (En microsegundos) entre el paquete actual y el paquete previo.
- *-xx* Imprime la información de los paquetes en hexadecimal
- *ether proto* \\ *ip and ip proto* *tcp* Realiza un filtro para mostrar la información únicamente de los paquetes Ethernet, IP y tcp-
- *-r ARCHIVO* Permite leer la información del archivo de entrada.

En la figura 4.1 se muestra la lectura de los datos utilizando la herramienta tcpdump.

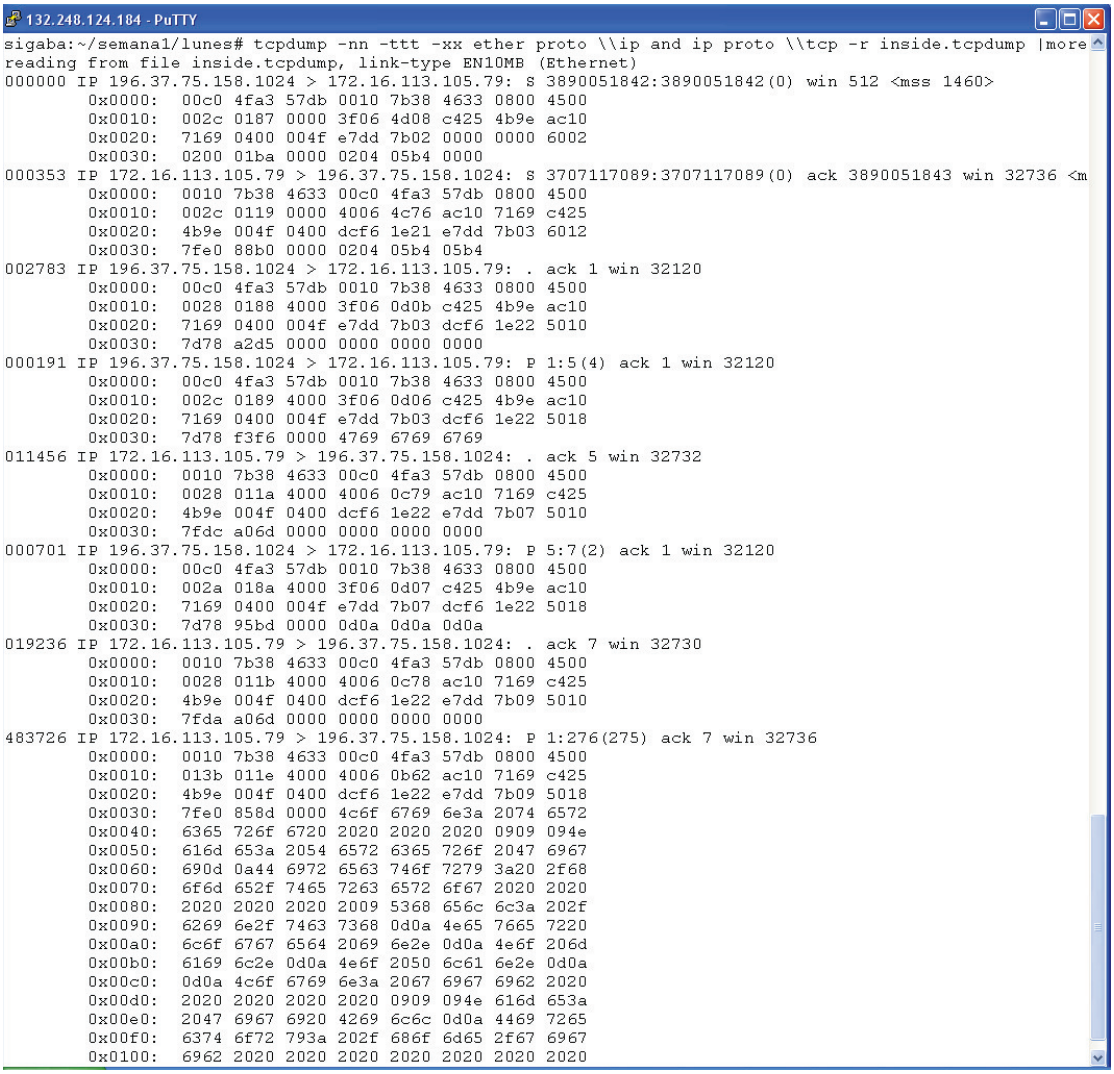

Figura 4.1: Lectura de Datos mediante TCPDUMP

4.5. Datos

Como se aprecia en la figura 4.1, es necesario usar un programa para el filtrado y traducción de los datos en un formato que pueda ser entendido por NuPIC, en este caso se generó un programa para hacer la traducción de la información anterior en un archivo de texto, que posteriormente se pasará a

al sensor de la Red Jerárquica.

Es importante especificar que información interesa filtrar, ya que de esto depende la construcción de la Red HTM, a continuación se muestra la información que es necesaria extraer.

4.5.1. Descripción de los datos

Al realizar la captura de todo el tráfico de la red, es necesario leer estos datos y analizarlos para obtener solo el tráfico que interesa analizar (Tráfico ethernet, ip y tcp), para lo cual se utilizó el analizador de protocolos de software libre *tcpdump*. Se tomo la decisión de utilizar esta herramienta ya que es la herramienta con la que se llevó acabo la captura de los datos, además permite leer los archivos como si fueran en tiempo real, para su análisis.

4.6. Topología

A continuación se muestra el modelo utilizado para la Red Jerárquica, donde se muestra el uso de 3 capas, la primera capa obtiene la información mostrada arriba, la segunda capa hace la clasificación del tipo de tráfico según sea el protocolo, y la ultima capa determina el tipo de tráfico. Esta ultima capa contiene el nodo que determina el tipo de tráfico analizado.

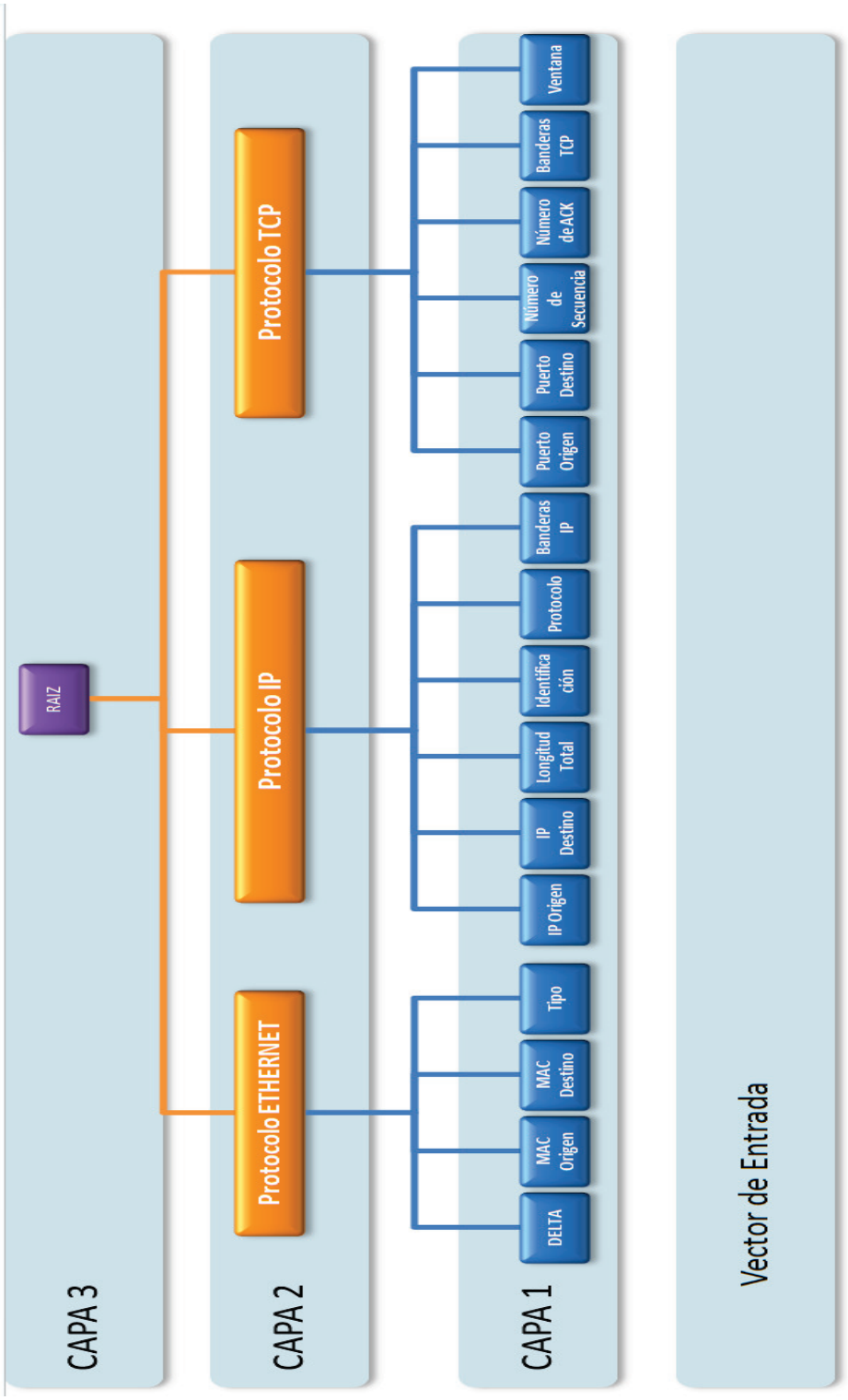

Figura 4.2: Red Jerárquica Temporal

Ya que la cantidad de tráfico de red es muy variada, es muy difícil contar con una categorización de los datos, por lo que no es posible realizar el aprendizaje supervisado, por tal motivo se realizará un aprendizaje no supervisado, es decir, no existirá una clasificación de los datos, con lo que la HTM creará las categorías y a su vez agrupará el los datos dependiendo de los datos de entrenamiento.

4.7. Vector del Sensor

Como se mencionó en la parte de arriba, se cuenta con 15 elementos que caracterizan a cada paquete de una red, además es necesario agregar la dependencia temporal de la cual se habló en el capitulo anterior, por lo que es necesario la diferencia de tiempo entre cada paquete, adicionalmente es necesario un identificador de cada paquete que permita de forma fácil buscar los paquetes considerados como anomalías, por lo que el vector necesita 17 elementos.

El vector del sensor debe estar formado por números enteros o flotantes, por lo que es necesario realizar la conversión de los datos, ya que algunos se encuentran en hexadecimal, como lo es en el caso de las direcciones MAC.

Las direcciones MAC están conformadas por 3 bytes que representan el OUI[6], y otros 3 bytes que representan el número asignado por el fabricante, actualmente se cuenta con un registro de 13132, pero el 96 % de dichos registros cuenta con el primer byte 0.

Debido a que al incrementar el número de elementos en el vector de sensor incrementa el tiempo de procesamiento, se descartó el 4 % de esos datos, es decir, únicamente se consideran todos aquellos paquetes que provengan de una dirección MAC que inicien con el primer byte 0, y dado que todos esos paquetes tienen por consecuencia un valor constante es posible removerlos, por lo tanto se consideran únicamente 5 bytes de las direcciones MAC origen y destino.

En la tabla 4.1 se muestra los campos contenidos dentro del Vector del Sensor.

[6]OUI es un acrónimo del inglés Organizationally Unique Identifier (Identificador Único Organizacional)

4.8. Aspectos a considerar

Es importante considerar que el filtrado y análisis de cada paquete requiere un tiempo de procesamiento, además del tiempo que se requiere para el proceso de aprendizaje y detección de posibles anomalías, por lo que la implementación del ids, no es en tiempo real, es decir, los paquetes que se reciben de la red, no son posibles analizarlos de forma rápida.

Debido a la gran cantidad de tráfico que circula por las redes, la implementación de este ids, no es aplicable para redes de gran tamaño, ya que el proceso de aprendizaje seria muy extenuante, por lo que si se desea implementar este IDS, se recomienda que sea bajo ambientes más controlados, como lo es en el caso de las Intranets, ya que teóricamente el tráfico que circula a través de estas es tráfico normal y las anomalías derivadas del uso de la red deberían ser por consecuencia pocas, con lo que es posible detectar anomalías y poder relacionarlas con posibles ataques.

El utilizar el IDS, en una red conectada a Internet, generaría muchos falsos positivos, ya que el proceso de aprendizaje seria bastante y al estar conectado con una red tan grande, es posible recibir cualquier tipo de paquetes que pudiera ser considerado anomalías.

Una vez realizado el proceso de detección de anomalías, no es posible determinar el tipo de amenaza a la que se esta expuesto, es decir, cuando el IDS detecta una anomalía, es necesario realizar un análisis de está para determinar la categoría de la amenaza, además se corre el riesgo de detectar un falso positivo.

Es primordial realizar una correcta etapa de entrenamiento de la red jerárquica, además de asegurarse que los datos de entrenamiento son los indicados así es decir, no se cuentan con posibles ataques, ya que el riesgo que sea realizado el entrenamiento con ataques, podría desencadenar que el IDS consideré que el ataque con el que se entreno, sea normal en la red y al presentarse otro ataque, esté no sea detectado.

Tabla 4.1: Campos del Vector del Sensor

ID	Delta_Time	MAC_Orig	MAC_Dest	Tipo	IP_Orig	IP_Dest
1	0.000000	16 123 56 70 51	192 79 163 87 219	2048	196 37 75 158	172 16 113 105
2	0.000346	192 79 163 87 219	16 123 56 70 51	2048	172 16 113 105	196 37 75 158
3	0.002833	16 123 56 70 51	192 79 163 87 219	2048	196 37 75 158	172 16 113 105
4	0.196045	192 79 163 87 219	16 123 56 70 51	2048	172 16 113 105	196 37 75 158
5	0.019837	16 123 56 70 51	192 79 163 87 219	2048	196 37 75 158	172 16 113 105
6	0.025788	16 123 56 70 51	192 79 163 87 219	2048	196 37 75 158	172 16 113 105
7	0.000347	192 79 163 87 219	16 123 56 70 51	2048	172 16 113 105	196 37 75 158
8	0.001029	16 123 56 70 51	192 79 163 87 219	2048	196 37 75 158	172 16 113 105
9	0.000264	192 79 163 87 219	16 123 56 70 51	2048	172 16 113 105	196 37 75 158
10	0.000890	16 123 56 70 51	192 79 163 87 219	2048	196 37 75 158	172 16 113 10

Long	Ident	Proto	Band_IP	P_Orig	P_Dest	Sec	ACK	Band_TCP	Vent
44	116	6	0	1024	25	2355892027	0	2	512
44	287	6	0	25	1024	927809984	2355892028	18	32736
40	117	6	64	1024	25	2355892028	927809985	16	32120
126	298	6	64	25	1024	927809985	2355892028	24	32736
40	122	6	64	1024	25	2355892028	927810071	16	32120
65	123	6	64	1024	25	2355892028	927810071	24	32120
66	299	6	64	25	1024	927810071	2355892053	24	32736
65	124	6	64	1024	25	2355892053	927810097	24	32120
87	300	6	64	25	1024	927810097	2355892078	24	32736
81	125	6	64	1024	25	2355892078	927810144	24	32120

Capítulo 5

Pruebas

5.1. Metodología de pruebas

Como se mencionó en el capitulo 3 el desarrollo de la red jerárquica realiza un proceso de aprendizaje no supervisado, es decir, no existe vector de categoría, por lo que el HTM crea las categorías para realizar la clasificación, dependiendo de los datos de entrada.

Como se realiza el proceso de entrenamiento y a través de tráfico normal, es decir, sin ataques, la red HTM clasificará cada paquete en varias categorías, posteriormente al haber aprendido el comportamiento normal de una red, cada nueva entrada en el proceso de pruebas, tratará de clasificarlo en las categorías previamente creadas, si la red HTM, no puede clasificar el vector de entrada, esta red creará una nueva categoría, estas nuevas categorías son consideradas anomalías de tráfico.

Posterior a la creación de nuevas categorías, corresponde analizarlo de forma manual para tratar de verificar si se trata de una nueva categoría a únicamente un nuevo comportamiento de tráfico, lo que no necesariamente se considera un ataque, esta es una etapa delicada, ya que la información presentada puede considerarse ataques.

El desarrollo de las pruebas, es decir, el filtrado, conversión de los datos de red y la ejecución del programa, se realizó en un servidor con las siguientes características:

- Marca: Dell
- Procesador: Intel Xeon a 1.86Ghz
- Memoria RAM: 2GB
- Disco Duro: 160GB
- Sistema Operativo: Linux Debian 5.0
- Versión NuPIC: 1.7.1

Es importante mencionar las características del equipo, debido a que la cantidad de tráfico analizado impacta en el rendimiento del procesamiento.

En las siguientes secciones se muestran las pruebas realizadas al IDS.

5.2. Datos Darpa

En esta sección se muestra la información de los datos obtenidos a las pruebas del IDS.

Haciendo referencia a los datos de la red de darpa, se cuenta con 5 semanas de tráfico, pero solo 2 de ellas corresponden al tráfico normal, es decir el tráfico sin ataques, esté trafico forma el perfil de red, este perfil se utiliza para el entrenamiento de la Red HTM.

Las otras 3 semanas, son realizadas con ataques, los cuales se puede observar a través de la pagina `http://www.ll.mit.edu/mission/communications/ist/files/master-listfile-condensed.txt`, esta lista es importante ya que a través de esta se puede observar el tráfico maliciosos que debe ser detectado por el IDS.

5.2.1. Etapa de Entrenamiento

Dentro de esta etapa de entrenamiento es necesario descargar el tráfico de las 5 semanas, posteriormente es necesario convertir dicho tráfico al formato establecido para el Vector del sensor, se tiene que tomar en cuenta que las capturas de tráfico se realizaron de Lunes a Viernes y se capturó el tráfico de salida y el de entrada, es decir, se cuentan con 50 archivos de capturas de tráfico.

Debido al gran tiempo de procesamiento y a la cantidad de pruebas, que se requería realizar, únicamente se tomo la semana 1 para realizar el entrenamiento y la semana 4 para las pruebas de detección.

A continuación se muestran los tiempos de entrenamiento de cada uno de las capturas de tráfico, el tiempo varia dependiendo del tamaño de los archivos.

El entrenamiento de la red HTM genera el archivo trained_IDS.xml, el cual contiene la salida mostrada a continuación:

```
more trained_IDS.xml
<?xml version="1.0" encoding="UTF-8"?>
<NumentaNet Version="1.1"
xmlns:nta="http://www.numenta.com">

<Node Name="Sensor">
<Class>VectorFileSensor</Class>
<Output Name="dataOut" ElementSize="4">
```

<ElementCount>31</ElementCount>
</Output>
<State></State>
<Property Name="Phase">
<Value> 0 <Value>
</Property>
</Node>
<Node Name="Level1[0]">
<Class>Zeta1Node</Class>
<Output Name="bottomUpOut" ElementSize="4">
<ElementCount>80001</ElementCount>
</Output>

Este archivo es la red HTM en formato xml, la cual es posible leer mediante NuPIC. Después de realizar el entrenamiento de la red, se genera el archivo trained_IDS.txt, el cual contiene la información de las categorías creadas como se muestra a continuación en la tabla 5.1.

Tabla 5.1: Salida red entrenada

Categoría	Identificador_de_paquete
20.4443	1
20.443	2
20.443	3
20.443	4
20.1446	5
20.1446	6
20.443	7
20.443	8
20.443	9
20.443	10
20.443	11
20.443	12
20.443	13
20.443	14
20.1446	15
20.1446	16
20.443	17

La columna de categoría muestra la categoría que la red HTM le asigno a cada paquete de entrenamiento, la siguiente columna corresponde al identificador de cada paquete, es importante considerar la categoría ya que en esta etapa la red clasificó todos los vectores en esas categorías, y por consecuente, todos los demás elementos que sean alegorizados de esa forma, serán considerados tráfico normal.

Antes de realizar la etapa de pruebas es necesario identificar los ataques que se realizaron, es decir los que se descargaron en la sección anterior, a continuación se muestra la tabla 5.2 con algunos ataques.

Tabla 5.2: Tráfico malicioso

Hora_inicio	Duración	IP_Destino	Ataque	Entrada/salida
08:18:35	00:04:07	172.016.112.050	ps	out
08:19:37	00:01:56	209.154.098.104	ps	out
08:29:27	00:00:43	172.016.112.050	ps	out
08:40:14	00:24:26	172.016.112.050	ps	out
08:48:12	00:00:02	172.016.114.050	sendmail	out
08:48:12	00:00:01	202.049.244.010	sendmail	out
09:15:05	00:00:01	172.016.113.050	portsweep	in
09:25:21	00:00:01	172.016.113.050	portsweep	in
09:35:36	00:00:01	172.016.113.050	portsweep	in
09:36:23	00:00:01	172.016.114.050	sshtrojan	out
09:36:23	00:00:01	202.077.162.213	sshtrojan	out
09:38:35	00:03:11	172.016.114.050	sshtrojan	out
09:39:37	00:00:31	135.013.216.191	sshtrojan	out
11:15:15	00:00:01	192.168.001.001	portsweep	in
11:16:15	00:00:01	192.168.001.001	portsweep	in
11:17:15	00:00:01	192.168.001.001	portsweep	in
11:18:15	00:00:01	192.168.001.001	portsweep	in
11:19:15	00:00:01	192.168.001.001	portsweep	in

En la primer columna de la tabla 5.2 se muestra la hora de inicio de cada ataque, en la segunda columna se observa el tiempo de duración del mismo, posteriormente se muestra la dirección IP Destino, después el nombre del Ataque y en la ultima columna, se tiene el tipo de tráfico del que se trata, es decir si es tráfico de entrada o de salida.

Como se mencionó anteriormente, es necesario acotar el tráfico que circula por una red de datos ya que la cantidad de tráfico es muy variado fue necesario acotarlo a Ethernet, IP y TCP, por lo que la información del tráfico UDP fue filtrada, y debido a que la información presentada en la colección de datos de DARPA incluye todo el tráfico y por consecuencia ataques sobre UDP, estos no serán identificados, ya que se descartaron desde el entrenamiento.

Esta pequeña colección de tráfico malicioso, va a servir como base de datos, donde se tendrán los ataques que se realizaron sobre la red, dicha base servirá para identificar los datos, es decir, si se crean nuevas categorías en la etapa de pruebas y en cualquiera de ellas se encuentra todo o parte del tráfico de la base se puede asegurar que se detecto una anomalía, la cual podrá ser ligada con la base para determinar el tipo de ataque, de lo contrario, si no se tuviera esta base solo se podría decir que se detecto una anomalía, pero no se podría decir que tipo de ataque es.

A continuación se muestra el tráfico del primer ataque, el cual se llevo acabó el día Lunes a las 08:18:35, este ataque recibió el nombre de ps, se puede consultar la descripción de cada ataque en la página `http://www.ll.mit.edu/mission/communications/ist/corpora/ideval/docs/attackDB.html`

/home/rsanchez/semana4/lunes# TZ=EST tcpdump r outside.tcpdump ether proto \\ip and ip proto \\tcp -nn| cat -n | more

6142 08:18:35.708492 IP 209.154.98.104.1027 > 172.16.112.50.23: S 284609283:284609283(0) win 512 <mss 1460>

6143 08:18:35.711103 IP 172.16.112.50.23 > 209.154.98.104.1027: S 342238218:342238218(0) ack 284609284 win 8760 <mss 1460>

6144 08:18:35.711463 IP 209.154.98.104.1027 > 172.16.112.50.23: . ack 1 win 32120

6145 08:18:35.718880 IP 209.154.98.104.1027 > 172.16.112.50.23: P 1:28(27) ack 1 win 32120

6146 08:18:35.765490 IP 172.16.112.50.23 > 209.154.98.104.1027: . ack 28 win 8760

6147 08:18:35.799578 IP 172.16.112.50.23 > 209.154.98.104.1027: P 1:16(15) ack 28 win 8760

6148 08:18:35.800061 IP 209.154.98.104.1027 > 172.16.112.50.23: P 28:40(12) ack 16 win 32120

6149 08:18:35.800814 IP 172.16.112.50.23 > 209.154.98.104.1027: P 16:31(15) ack 40 win 8760

6150 08:18:35.815186 IP 209.154.98.104.1027 > 172.16.112.50.23: . ack 31 win 32120

6151 08:18:35.815873 IP 172.16.112.50.23 > 209.154.98.104.1027: P 31:52(21) ack 40 win 8760

6152 08:18:35.817074 IP 209.154.98.104.1027 4 > 172.16.112.50.23: P 40:134(94) ack 52 win 32120

6153 08:18:35.820160 IP 172.16.112.50.23 > 209.154.98.104.1027: P 52:101(49) ack 134 win 8760

6154 08:18:35.835190 IP 209.154.98.104.1027 > 172.16.112.50.23: . ack 101 win 32120

6155 08:18:35.870578 IP 172.16.112.50.23 > 209.154.98.104.1027: P 101:107(6) ack 134 win 8760

6156 08:18:35.871017 IP 209.154.98.104.1027 > 172.16.112.50.23: P 134:140(6) ack 107 win 32120

6157 08:18:35.871744 IP 172.16.112.50.23 > 209.154.98.104.1027: P 107:114(7) ack 140 win 8760

6158 08:18:35.885185 IP 209.154.98.104.1027 > 172.16.112.50.23: . ack 114 win 32120
6159 08:18:35.885848 IP 172.16.112.50.23 > 209.154.98.104.1027: P 114:117(3) ack 140 win 8760
6160 08:18:35.905184 IP 209.154.98.104.1027 > 172.16.112.50.23: . ack 117 win 32120
6161 08:18:36.805848 IP 209.154.98.104.1027 > 172.16.112.50.23: P 140:141(1) ack 117 win 32120
6162 08:18:36.806630 IP 172.16.112.50.23 > 209.154.98.104.1027: P 117:118(1) ack 141 win 8760

La opción TZ=EST, corresponde al uso horario en el que se recolectaron los datos, en este caso al Este de Estados Unidos, la opción cat -n cuenta el numero de lineas, para nuestro caso, es el numero de identificador asociado a ese paquete, este número se utilizará para detectar las anomalías.

A continuación se muestra en la tabla 5.3 los tiempos necesarios para el entrenamiento de la semana 1 y los de prueba de la semana 4, los datos que se presentan estan dados en horas.

Tabla 5.3: Tiempos de realización de pruebas

Etapa	Lunes	Martes	Miércoles	Jueves	Viernes
Entrenamiento	34.8	33.1	47.1	67.2	32.5
Pruebas	28.1	27.4	23.0	24.8	22.1

El largo tiempo en la realización de las pruebas se debe al tamaño de los paquetes capturados, es decir, cada día de captura generó cerca de un millón y medio de vectores, por ejemplo el día Jueves en la etapa de entrenamiento, el tamaño generado fue cercano a los 3 millones de vectores.

5.2.2. Etapa de Pruebas

Una vez realizada la etapa de entrenamiento, se realizó la etapa de pruebas, repitiendo los pasos anteriores, crear un vector del sensor con la información del tráfico de prueba, este tráfico contiene tanto información de tráfico normal como anormal, despues se ejecutó la herramienta, donde se optubieron.

A continuación se muestra la tabla 5.4 el resultado de la prueba, el cual originó el archivo test_IDS.txt, este archivo es similar al trained_IDS.txt.

Tabla 5.4: Salida datos de prueba

Categoría	Identificador_de_paquete
8037.88	1
7919.08	2
7942.59	3
7989.16	4
8059.14	5
8037.82	6
7919.08	7
7919.08	8
7919.08	9
7919.08	10
7919.08	11
7919.08	12
7919.08	13
7919.08	14
8037.82	15
20.1446	16

La diferencia más significativa es la generación de nuevas categorías, es importante no confundir el identificador de paquete con el anterior, este identificador de paquete corresponde a los datos de prueba.

Como se mencionó anteriormente la red HTM generó categorías y clasificó los datos en estás durante la etapa de entrenamiento, por lo que al momento de realizar la etapa de pruebas tratará de clasificar el nuevo tráfico en las categorías previamente creadas, en caso de no poder realizar dicha clasificación la red HTM generará nuevas categorías, si esto sucediese se puede

decir que se generó una anomalía. Dicho lo anterior, se generó una lista de las categorías creadas en el entrenamiento, y se comparó con la lista de las categorías generadas en la etapa de pruebas, las cuales se muestran en la tabla 5.5.

Tabla 5.5: Categorías generadas

Categoría_Entrenamiento	Categoría_Pruebas
0.153	0.128614
0.153001	0.128615
0.153002	0.128617
0.153004	0.128618
0.153247	0.128738
0.153248	0.128739
0.153249	0.128857
0.153703	0.128863
0.182583	0.128865
0.202267	0.128981

El siguiente procedimiento es buscar si el Identificador de paquete de la base de datos de tráfico malicioso se encuentra en una categoría nueva del archivo, sí se encuentra en una nueva categoría se puede decir que se generó una anomalía de tráfico, por lo que detectó el tráfico malicioso. A continuación se muestra en la tabla 5.6 los ataques que fue posible detectar.

Tabla 5.6: Ataques detectados del día Lunes 4a semana

Tipo de Ataque	Detectado
ftpwrite	No
guesstelnet	No
portsweep	Si
ps	Si
secret	No
sendmail	Si
smurf	No
snmpget	No
sshtrojan	Si
xsnoop	Si

5.3. Datos de Red Real

Los datos mostrados en la sección anterior aplican a un escenario de pruebas, es decir, un escenario en el cual se proporcionan datos capturados, que son proporcionados por la evaluación de DARPA.

Utilizando la red de datos del proyecto Honeynet UNAM se realizaron capturas en el tráfico de red, además se solicitó la realización de ataques controlados sobre equipos en la red, con la finalidad de utilizar el IDS, en una red de datos real, mostrando el funcionamiento sobre un ambiente en operación.

El desarrollo de la red HTM es exactamente el mismo ya que se diseñó para identificar patrones de tráfico malicioso, únicamente variarán tanto los datos de pruebas como los de entrenamiento, con la diferencia que al tratarse de una red real, se pueden encontrar variaciones en los resultados.

Cabe recordar que el diseño de esta red HTM esta acotado únicamente a las Intranets, es decir, redes que no cuentan con acceso a internet, ya que la cantidad de tráfico que se debería analizar y entrenar seria bastante.

La cantidad de tráfico malicioso fue relativamente pequeño, esto debido a las implicaciones, es decir, fue difícil ejecutar ataques hacia equipos ya que se trataba de una red en producción, lo que implica que podría afectar la operación, debido a esta limitante, únicamente fue posible lanzar 15 ataques para realizar las pruebas, se decidió implementar el esquema anterior, es decir, se realizaron capturas de tráfico de una semana sin ataques y posteriormente se realizaron ataques sobre la red.

A continuación se muestra en la tabla 5.7 los tiempos en lo que se llevaron acabo las pruebas sobre la red real, las cantidades estan dadas en horas.

Tabla 5.7: Tiempos de realización de pruebas sobre la red de datos real

Etapa	Lunes	Martes	Miércoles	Jueves	Viernes
Entrenamiento	18.1	18.2	17.6	19.2	15.8
Pruebas	13.1	10.7	10.5	10.9	10.8

5.3.1. Etapa de Entrenamiento

En esta etapa de entrenamiento fue necesario colocar un PortMirror que ya se mencionó en el tercer capítulo, esto con la finalidad de realizar la captura de todo el tráfico que pasa por la red.

Una vez establecido el PortMirror en el Switch, es necesario colocar un Sniffer que permita capturar el tráfico, se decidió utilizar tcpdump, ya que las capturas de las pruebas anteriores se utilizó la misma herramienta y se pretende que la prueba varié únicamente en la información a analizar y a entrenar.

Se repiten los pasos anteriores para generar el vector del sensor para entrada y salida.

En esta ocasión se realizaron los ataques sobre la red utilizando la herramienta Metasploit, la cual contiene una serie de paquetes que permite la realización de ataques sobre equipos de cómputo.

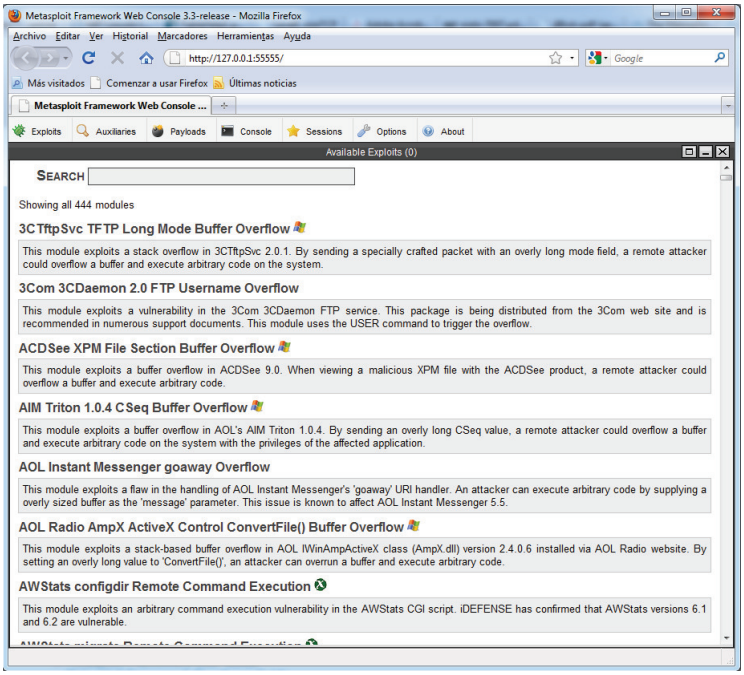

Figura 5.1: Herramienta para generar ataques, Metasploit

Se realizaron los mismos procedimientos anteriores, esta ocasión únicamente se realizaron los siguientes ataques:

- Fuerza Bruta
- Escaneo de puertos
- sql-injection

- Cross Site Scripting

5.3.2. Etapa de Pruebas

Una vez realizada la etapa de entrenamiento, es necesario repetir los mismos pasos que que se llevaron acabo para los datos de Darpa, después de realizar el entrenamiento únicamente fue posible detectar los ataques mostrados en la tabla 5.8.

Tabla 5.8: Ataques detectados en una red real

Ataque	Detectado
Fuerza Bruta	Si
Escaneo de puertos	No
Sql-Injection	Si
Cross Site Scripting	No

A continuación se muestran las conclusiones generales del desarrollo de esta tesis.

Capítulo 6

Conclusiones

6.1. Conclusiones y Trabajo futuro

Como se muestra en el capítulo 5 fue posible detectar ataques utilizando HTM, cumpliendo con el objetivo de esta tesis. La detección de anomalías utilizando Nupic tiene limitaciones como el tiempo que tardan los datos en ser procesados, esta limitación hace imposible realizar un IDS en tiempo real y por consecuencia imposibilita colocarlo como un dispositivo en línea. La colocación en modo pasivo sí es posible, es decir el colocarlo en una red para recibir la información sin alterar el flujo. Las pruebas realizadas tanto a los datos de DARPA como a los datos de la red en producción se obtuvieron mediante capturas utilizando la herramienta tcpdump, para posteriormente ser analizados.

La mayor problemática encontrada en la realización de este trabajo de tesis fueron sin lugar a duda los *falsos positivos*. Estos eventos se detectaron al ser considerados anomalías en el tráfico de red (Cabe aclarar que una anomalía en el tráfico de red implica un evento del cual no se tenía registro alguno).

En el capítulo 2 se describieron las ventajas y desventajas de los Sistemas de Detección de Intrusos tanto basados en anomalías como basados en firmas, las cuales fueron tomadas en consideración para el desarrollo de esta tesis. Se generaron falsos positivos en la etapa de pruebas debido a que el periodo de entrenamiento abarcó grandes cantidades de tráfico, siendo una muestra significativa del comportamiento de una red de datos. Al momento de generar un nuevo patrón en el tráfico de red, se puede esperar que sea levantada una anomalía en el sistema, la cual no necesariamente es un ataque.

La consideración de la que se acaba de hablar es una de las mayores problemáticas de los IDS basados en anomalías, este problema puede ser corregido mediante la generación de perfiles que describan el tráfico de red, dichos perfiles deberían cambiar en el tiempo con respecto a las necesidades y uso de cada red.

Dentro del trabajo futuro en este proyecto, se sugiere nuevos modelos de red HTM para generar diferentes aprendizajes. A pesar que los resultados obtenidos en el trabajo de esta tesis son aceptables, ya que se pudieron detectar ataques de red basados en anomalías, pueden mejorarse por mucho para igualar la efectividad de otros sistemas comerciales, además los valores establecidos utilizados dentro de NuPIC, fueron valores por omisión que recomienda Numenta. Adicional al proceso de aprendizaje e inferencia, se debe considerar, el tratamiento “paquete por paquete”, es decir, el poder analizar

cada paquete de tráfico y determinar si es o no malicioso en el momento en que se reciba dicho paquete, este proceso ayudará a la colocación de esta herramienta en un escenario en tiempo real.

Se les recomienda a las personas que deseen continuar con el desarrollo o implementación de Sistemas de Detección de Intrusión utilizando NuPIC para la detección de anomalías, que establezcan escenarios controlados como es el caso de las Intranets, ya que el tráfico que viaja a través de esas redes es tráfico sin anomalías, con lo cual es posible establecer un perfil adecuado para analizar.

Les agradezco profundamente a las personas que leyeron este trabajo de tesis, esperando les apoye en trabajos de investigación o consulta.

Apéndice A

Anexo

A.1. Anexo

El código del programa se encuentra disponible en la siguiente dirección: `http://sigaba.seguridad.unam.mx:8080/ids.tar.gz`

Glosario de Acrónimos

ACK Acknowledgement mejor conocido como acuse de recibo

BBS Bulletin Board System o Sistema de Talón de Anuncios

Bitenet Antigua red internacional de computadoras de centros docentes y de investigación que ofrecía servicios interactivos de correo electrónico y de transferencia de archivos.

CERT Computer Emergency Response Team

DARPA Defense Advanced Research Projects Agency o Agencia de Investigación de Proyectos Avanzados de Defensa

DMZ Zona Desmilitarizada

HTM Hierarchical Temporal Memory

IDS Intrusión Detection System o Sistema de Detección de Intrusos

IEEE Institute of Electrical and Electronics Engineers

LAN Local Area Network o Red de Área Local

MAC Media Access Control o Control de Acceso al Medio

NUPIC Numenta Platform for Intelligent Computing o Plataforma Numenta para Cómputo Inteligente

TCP Transmission Control Protocol o Protocolo de Control de Transmisión

UDP User Datagram Protocol o Protocolo de Datagrama de Usuario

VLAN Virtual LAN, red de área local virtual, método de crear redes lógicamente independientes dentro de una misma red.

Bibliografía

[1] Anderson, J., Computer Security Threat Monitoring and Surveillance, National Institute of Standards and Technology, Fort Washington, 1980.

[2] Barlet, P., H. Pujol, J. Barrantes, J. Sóle y J. Domingo, *A System for Detecting Network Anomalies based on Traffic Monitoring and Prediction*, RedIris, Jornadas Técnicas RedIris, 2005.

[3] Dileep, G., *How the brain might work*, A Hierarchical and temporal model for learning and recognition, Thesis for the degree of doctor of philosophy, Stanford University, 2008.

[4] Dileep, G. y J. Hawkins, *Hierarchical Temporal Memory, Concepts, Theory, and Terminology*. Numenta Inc., pp. 1-12, 2006.

[5] Dileep, G. y B. Jaros, *The HTM Learn-ing Algorithms*, Numeta Inc, 2007, `http://www.numenta.com/for-developers/education/Numenta_HTM_Learning_Algos.pdf`.

[6] Gerod, M., *Using Hierarchical Temporal Memory for Detecting Anomalous Network Activity*, Thesis for Degree of Master of Science, Air Force Institute of Technology, 2008.

[7] Hawkins, J. y S. Blakeslee, *On Intelligence*, Times Books, 2004.

[8] Haines, J., L. Rossey, R. Lippmann and R. Cunnigham, *Extending the 1999 Evaluation, In the Proceedings of DISCEX 2001*, Massachusetts Institute of Techonology, Lincoln Laboratory Publications, 2001, `http://www.ll.mit.edu/mission/communications/ist/files/discex01_paper.pdf`.

[9] Haines, J., P. Richard, R. Lippmann, J. Fried, E. Tran, S. Boswell, and A. Zissman, *1999 DARPA Intrusion Detection System Evaluation: Design and Procedures*, Massachusetts Institute of Technology, Lincoln Laboratory Publications, 2000, `http://www.ll.mit.edu/mission/communications/ist/files/TR-1062.pdf`.

[10] IEEE, *Organizationally Unique Identifie*, IEEE Standards Association, 2009, `http://standards.ieee.org/regauth/oui/index.shtml`.

[11] Jacob, W. y J. E. Gaffney, *Evaluation of Intrusion Detection Systems*, Jurnal of Research of the National Institute of Standards and Technology, Volume 108, Number 6, 2003, `http://nvl.nist.gov/pub/nistpubs/jres/108/6/j86ulv.pdf`.

[12] Javitz, S. and A. Valdes, *The SRI IDES Statistical Anomaly Detector*, IEEE, 1991, `http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=130799&isnumber=3628`.

[13] Jones, D., *Network Management for the Mid-Market*, Series Editor, 2007.

[14] Lin, Y., C. Jan, and P. Lin, *Designing an Integrated Architecture for Network Content Security Gateways*, National Chiao-Tung University and National Taiwan University of Science and Technology, 2006.

[15] Lippmann, P. and J. Haines, *Analysis and Results of the 1999 DARPA Off-Line Intrusion Detection Evaluation, in Recent Advances in Intrusion Detection*, Third International Workshop, RAID 2000 Toulouse, France, October 2000.

[16] Massicotte, F., M. Couture, L. Briand and Y. Labiche, *Context-Based Intrusion Detection Using Snort, Nessus and Bugtraq Databases*, Department of Systems and Computer Engineering, Carleton University, 2008.

[17] Mell, P., P. Hu, R. Lippmann, J. Haines and M. Zissman, *An Overview of Issues in Testing Intrusion Detection Systems*, National Institute of Standards and Technology ITL, y Institute of Technology Lincoln Laboratory, 2003, `http://csrc.nist.gov/publications/nistir/nistir-7007.pdf`.

[18] Nathan, C., *Song Identification using Numenta Platform for intelligent Computing*, Ohio State University, 2008.

[19] Net Optics, *Deploying Network Taps with Intrusion Detection Systems*, Net Optics Inc, 2004, `www.netoptics.com/products/pdf/Taps-and-IDSs.pdf`.

[20] Parthasarathy, M., *Analysis of network management of remote network elements*, IEEE, International Conference on Systems and International Conference on Mobile Communications and Learning Technologies, 2006.

[21] Patterson, D., M. Howell and M. Digonnet, *Noninvasive Switchable Acousto-Optic Taps for Optical Fiber*, IEEE, Journal of Lightwave Technology, Volume 8, 1990.

[22] Roesch, M., *Snort - Light weight intrusion detection*, Usenix, Washington, 1999, `http://www.usenix.org/event/lisa99/full_papers/roesch/roesch.pdf`.

[23] Samaha, S., *An Intrusion Detection System for the Air Force*, Proceedings of the Fourth Aurospace Computer Security Applications Conference, Orlando, FL, December 1988.

[24] Scarfone, K. y P. Mell, *Guide to Intrusion Detection and Prevention Systems*, National Institute of Standar and Tecnology, Special Publication 800-94, 2007, `http://csrc.nist.gov/publications/nistpubs/800-94/SP800-94.pdf`.

[25] Siamwalla, R., R. Sharma and S. Keshav, *Discovering Internet Topology*, Cornell Network Research Group Department of Computer Science, Cornell University, 2009.

[26] Stallings, W., *Network Security Essentials, Aplications, and Standars*, Prentice Hall, Segunda Edición, pp. 5-19, 2009.

[27] Tanembaum, A., *Redes de Computadoras*, Cuarta Edición, Prentice Hall, 2003.

[28] Vaccaro, H. and G. E. Liepins, *Detection of Anomalous Computer Session Activity*. IEEE, Symposium on Security and Privacy, Oakland, CA, 1989.

[29] Vigna J. and A. Kemmerer, *NetSTAT: A Network-based Intrusion Detection Approach*, Reliable Software Group, Department of Computer Science, University of California Santa Barbara, 1998.

[30] Villalón, A., *Seguridad en Unix y en Redes*, TLDP-E, pp. 1-15, 2002.