Tesis de Maestría en Redes de Datos.
Detección de Intrusiones en redes de datos con captura distribuida y procesamiento estadístico

Autor: Britos José Daniel

Director: Javier Díaz

1 de septiembre de 2010

Índice general

Capítulo 1

Introducción

1.1. Motivación y Estado del Arte de los IDS

El crecimiento de las aplicaciones de comercio electrónico basado en Internet y los ataques a las redes se han vuelto más comunes y sofisticados.

Las redes demandan medidas de protección más elaboradas para garantizar una segura operación y dar continuidad a los servicios críticos. Estas medidas exigen e incluyen métodos de detección y repuesta a los intentos de intrusión en tiempo real.

En el presente trabajo se proponen modelos estadísticos y clasificadores multivariables para detectar perfiles de tráfico anómalos, como así también el uso de clasificadores basados en redes neuronales o en agentes cooperantes.

La preocupación por la seguridad en redes y servidores se inició cuando comenzaron a interconectarse las computadoras entre sí, Anderson[1], en su temprano trabajo que presenta en el año 1972 expone los principales ataques a lo sistemas informáticos, los que con el transcurrir de las décadas se han perfeccionado y complejizado.

En el año 1983 el DoD (Department of Defence USA) publica una de las primeras y más difundidas normas de seguridad, y entre ellas la versión del año 1985 [2] la que aún está en vigencia hoy.

En el año 1995 El British Standards Institute (BSI) publica la norma de seguridad BS 7799 Part

1 que finalmente deriva en el conjunto de normas ISO 27000 [3], sin embargo las mismas están más orientadas a la seguridad de los sitios que a la de las comunicaciones.

La Unión Internacional de Comunicaciones (ITU) en el año 1991 elabora la normativa para la seguridad de las comunicaciones ITU X800 [4] y ITU X805 [5].

En el 2000 la Internet Society network working group elabora un glosario de seguridad informática [6]; todo este conjunto de normas continúa en pleno desarrollo, y la ISO posee un calendario de normas de seguridad hasta el presente.

La aplicación de las recomendaciones de las normas no fue tarea suficiente para proteger las redes, sino que para ello debieron implementarse barreras activas como firewalls [7] [8]. Desde el año 1980 los firewall logran proteger a la red de una gran variedad de ataques, sin embargo existen otros ataques que se presentan ante los firewall con apariencia de tráfico normal.

Mediante la simple inspección de los paquetes es muy difícil descubrir si se ha tratado de un tráfico normal o de uno malicioso, por lo que se hace necesario recurrir a sistemas de detección más específicos, que analicen los paquetes que entran a la red, elaboren estadísticas y traten de identificar el tráfico malicioso; estos sistemas se denominan IDS (Intrusion Detection System). [9].

El concepto de IDS fue introducido en el año 1980 por J. P. Anderson[10] y desde ese momento se constituyó en el foco de una amplia tarea de investigación.

Existen una variedad de IDS, tales como los basados en Servidores llamados HIDS (Host Intrusion Detection System) y en Sistemas de Archivos y Redes, llamados NIDS (Network Intrusion Detection System).

Hasta el presente se han desarrollado numerosos y diversos NIDS, entre los cuales de destaca el SNORT [11], como un programa detector de Intrusiones más difundido en el ámbito de la seguridad en redes; esta herramienta se distribuye bajo licencia GNU, y recibe el aporte de investigadores de todo el mundo. Snort tiene módulos bajo licencia GNU y módulos propietarios, se puede considerar a Snort como una plataforma, para experimentar con módulos de detección sin tener que preocuparse de la captura de los datos.

Otro desarrollo importante en NIDS es el Bro [12] que provee una plataforma para experimen-

tación, detección y estudio de ataques a redes, y por otro lado permite la captura distribuida de los paquetes que atraviesan la red.

A pesar de que existe un sin número de programas para la detección de intrusiones, es mayor todavía el vacío de soluciones para el grande y diverso tráfico malicioso. Esto es así porque el avance en el poder de cálculo de los microprocesadores modernos, permitió elaborar estrategias y algoritmos que eran prohibitivos hasta hace algunos años, como los complejos tratamientos estadísticos y el uso de redes neuronales.

Un antecedente en este tema es el trabajo de Papavassiliou [13] quien propuso el método de detección de intrusiones estadísticas, utilizó como herramienta la estadística de Kolmogorov-Smirnov junto a redes neuronales para modelar y detectar ataques. Podríamos decir que todo su trabajo se basó en simulaciones en computadoras, no llegando a probarlo en un ambiente real.

Otro hallazgo interesante y nodal lo ofreció la observación y estudio acerca de la metodología de autorganización en la inteligencia colectiva que poseen las colonias de hormigas, y su aplicación en el análisis del tráfico de redes [14] [15]. Si bien este mismo método ha sido implementado por otros autores, no es menos valioso el aporte de Gopalakrishna [16] cuando incorpora una captura distribuida de datos.

Los IDS y particularmente los métodos estadísticos, el uso de redes neuronales y los sistemas de autoorganización de inteligencia colectiva o agentes cooperantes son los temas más investigados actualmente, descriptos en una profusa publicación de artículos.

1.2. Contribuciones

La vulnerabilidad de las redes de datos está presente, principalmente, en los campus universitarios y en las grandes empresas que poseen más de un punto de acceso a Internet, redes que por su tamaño y diversidad de servicios prestados son un blanco de virus, gusanos y hackers.

En estos casos, los antivirus y los firewall no son suficientes defensa y no aseguran una protección eficaz. Debido a ello es necesario proveer a las redes con sistemas de protección bien planeados y políticas integrales de defensa contra los ataques.

Dentro de este ámbito, juegan un papel importante los sistemas de alerta temprana, y para ello es imprescindible la utilización de Sistemas de Detección de Intrusiones en Redes (NIDS).
El propósito de este trabajo es investigar nuevos métodos, los que utilizados en otras áreas de la ciencia han enfrentado problemáticas similares para aplicarlos con eficacia en la detección de intrusiones en redes de datos.
En este sentido, el proceso que sigue la detección de intrusiones involucra etapas tales como, la captura de los datos, la selección estadísticas de los más relevantes, hasta llegar al mecanismo de decisión que detecta a un ataque.
En cada una de estas etapas se pueden aplicar nuevas técnicas para perfeccionar el estado del arte de los IDS actuales.

1.3. Objetivo de esta Investigación

El enfoque de este estudio se orienta al análisis y desarrollo de tecnologías basadas en la investigación estadística, las redes neuronales y los sistemas autónomos aplicados a los problemas de detección de intrusiones en redes de datos.
A lo largo de su desarrollo se pretende consolidar mejores métodos para detectar dichos ataques, para lo cual se seleccionan los más apropiados elementos de juicio que hagan efectivos y óptimos los métodos de defensa.
Los objetivos específicos de este trabajo se sumarizan en el siguiente orden:

- Proponer una arquitectura realista y bien estructurada de los métodos de defensa, a los fines de ser implementados en cualquier sitio.
- Demostrar y comprobar paso a paso, las hipótesis y las propuestas teóricas mediante el análisis de los datos tomados de la realidad.
- Poner de manifiesto el dominio en el conocimiento de la seguridad informática y de los IDS, de tal forma que ellos constituyan el ítem inteligente en la elección de los algoritmos

apropiados, cuestión de evitar la incumbencia de un problema en algún algoritmo, en particular.

- Implementar un prototipo de los algoritmos propuestos.

1.4. Lineamiento de la Tesis

En este primer capítulo se expuso las necesidades de la defensa de las redes, los problemas a los que se enfrenta el sistema y las contribuciones que se proponen para solucionarlos, mediante un prototipo desarrollado.

En el capítulo 2 se presentan los ataques y vulnerabilidades más comunes encontradas en las redes.

En el capítulo 3 se realiza una introducción al estudio de los aspectos más relevantes de los IDS.

El capítulo 4 aborda los métodos estadísticos para realizar la selección de los datos y su clasificación.

En el capítulo 5 se describen las redes neuronales y su aplicación a los IDS.

En el capítulo 6 se desarrolla la inspección profunda de paquetes.

En el capítulo 7 se presentan los IDS Distribuidos y la técnica de colonia de hormigas.

En el capítulo 8 se describen las herramientas de ataques utilizadas para la evaluación de los métodos presentados.

En el capítulo 9 se exponen las conclusiones y proponen lineas de investigación para la continuación en las investigaciones en IDS.

Capítulo 2

Ataques y Vulnerabilidades

2.1. Seguridad de la Información el Paradigma C.I.A.

La información es un activo esencial para las operaciones de cualquier organización, y por lo tanto necesita ser protegida convenientemente. La seguridad de la información [17] es una disciplina que tiene por objeto asegurar y proteger las tres propiedades fundamentales de la información de los sistemas:

- Confidencialidad: Es la habilidad de un sistema para presentar sus recursos accesibles solo a las partes autorizadas a su uso.
- Integridad: Es la habilidad de un sistema que permite que solo las partes autorizadas puedan modificarlo y solo en las formas que son consistentes con las funciones realizadas por el sistema.
- Disponibilidad: Los derechos válidos de acceso a la información nunca deben ser denegados y deben ser satisfechos en tiempo y forma.

Estos paradigmas son conocidos como C. I. A. Algunos autores agregan paradigmas como autenticación, no repudiación, seguridad. Sin embargo existe un amplio consenso que todos los demás pueden ser derivados de los tres paradigmas básicos.

Hoy en día la mayor parte de la información en uso es procesada a través de sistemas de computación, por esto es común que el termino “Seguridad de la Información” se use para denotar “Seguridad de Computadoras”, pero académicamente hablando “Seguridad de la Información” se extiende a todos los procesos de manejo y almacenamiento de la información ya sea en papel o almacenada electrónicamente ya sea enviada por vía postal o usando medios electrónicos.
El “U. S. National Information System Security Glossary” define La seguridad de los sistemas de información (INFOSEC) [18] como:

> “la protección de los sistemas de información contra el acceso o la modificación sin autorización ya sea estén almacenados, procesados o en transito, y contra la denegación de servicio a los usuarios debidamente autorizados, incluyendo las medidas necesarias para detectar, documentar y contrarrestar tales intentos”.

2.2. El paradigma A. A. A.

El paradigma de Confidencialidad, Integridad y Disponibilidad de la información contenida en un sistema de computadoras es usualmente implementado a través de la arquitectura A. A. A.:

- Autenticación: El usuario es propiamente identificado de alguna manera y un perfil de acceso es asociado a él.
- Autorización: Cada operación y tarea activada por el usuario esta sujeta a un conjunto de restricciones, dadas por los privilegios de acceso a los activos del sistema
- Auditabilidad: Las operaciones y las tareas realizadas son registradas con un proceso propio en orden a asegurar que no se ha producido ninguna violación a los paradigmas de la C. I. A.

Este concepto de taxonomía A.A.A. se aplica a los sistemas operativos de red y a los servicios de red, pero también a los sistemas de control de las redes, tales como firewalls, VPN.

Esto sucede porque la idea de autorización y autenticación son ortogonales a la mayoría de los procesos y servicios de red. La autenticación puede ser realizada a través de varias técnicas, a menudo divididas en las siguientes [19]:

- Algo que el usuario debe conocer por ejemplo la palabra clave.
- Algo que el usuario debe poseer por ejemplo “tarjetas inteligentes”, “llaves”.
- Algo que el usuario es por ejemplo huella digital, iris del ojo.

Diferentes modelos han sido propuestos en la literatura para la gestión del control de acceso en aplicaciones distribuidas. Tradicionalmente, los modelos de control de acceso han sido caracterizados mediante modelos DAC (Discretionary Access Control) y modelos MAC (Mandatory Access Control). Posteriormente modelos RBAC (Role-Based Access Control) o modelos TBAC (Task-based access control) han sido propuestos para gestionar los requerimientos de seguridad en un gran conjunto de aplicaciones. A continuación se resumen las características de estos modelos junto con sus limitaciones más importantes [20].

2.2.1. Modelo DAC

El modelo de control de acceso discrecional (DAC, Discretionary Access Control), también llamado modelo de seguridad limitada, es un modelo no orientado al control del flujo de sistema son controlados y se especifican reglas de autorización de acceso para cada sujeto y objeto. Los sujetos pueden ser usuarios, grupos o procesos. Los modelos DAC están basados en la idea de que el propietario de un objeto, su autor, tiene el control sobre los permisos del objeto. Es decir, el autor es autorizado a permitir u otorgar permisos para este objeto a otros usuarios. DAC admite la copia de datos desde un objeto a otro por usuarios autorizados de manera que un usuario puede permitir el acceso para copiar datos a otro usuario no autorizado. Este riesgo pude ser extendido a todo el sistema violando un conjunto de objetos de seguridad. La principal ventaja de DAC es que el usuario se beneficia de la flexibilidad del modelo. Sin embargo es difícil para DAC garantizar las reglas de integridad como “least privilege” o “separation of

duty" que son necesarias en los ambientes con procesos colaborativos. DAC es apropiado en ambientes donde la compartición de información es más importante que su protección.

2.2.2. Modelo MAC

En el modelo de control de acceso obligatorio (MAC, Mandatory Access Control) todos los sujetos y objetos son clasificados basándose en niveles predefinidos de seguridad que son usados en el proceso de obtención de los permisos de acceso. Para describir estos niveles de seguridad todos los sujetos y objetos son marcados con etiquetas de seguridad que siguen el modelo de clasificación de la información militar (desde "desclasificado" hasta "alto secreto"), formando lo que se conoce como política de seguridad multinivel. Por este motivo se define MAC como un modelo "multinivel" [21]

2.2.3. Modelo RBAC

El principal objetivo del modelo de control de acceso basado en rol (RBAC, Role Based Access Control) es prevenir que los usuarios tengan libre acceso a la información de la organización. [22]. El modelo introduce el concepto de rol y asocia a los usuarios con los roles por los que va pasando durante la vida del sistema. Los permisos de acceso están asociados a los roles. El rol es un concepto típico usado en empresas para ordenar y estructurar sus actividades organizativas. RBAC permite modelar la seguridad desde de una perspectiva empresarial puesto que podemos conectar los requerimientos de seguridad con los roles y las responsabilidades existentes en la organización. RBAC está basado en la definición de un conjunto de elementos y de relaciones entre ellos. A nivel general describe un grupo de usuarios que pueden estar actuando bajo un conjunto de roles y realizando operaciones en las que utilizan un conjunto de objetos como recursos. En una organización, un rol puede ser definido como una función que describe la autoridad y responsabilidad dada a un usuario en un instante determinado. Entre estos cuatro elementos se establecen relaciones del tipo:

- Relaciones entre usuario y roles, modelando los diferentes roles que puede adoptar un

usuario.

- Conjunto de operaciones que se pueden realizar sobre cada uno de los objetos. A los elementos de esta relación se les denomina permisos.

- Relaciones entre los permisos y los roles. Modelamos cuándo un usuario, por estar en un rol determinado, tiene permiso para realizar una operación sobre un objeto.

El modelo RBAC [23] incluye un conjunto de sesiones donde cada sesión es la relación entre un usuario y un subconjunto de roles que son activados en el momento de establecer dicha sesión. Cada sesión esta asociada con un único usuario. Mientras que un usuario puede tener una o más sesiones asociadas. Los permisos disponibles para un usuario son el conjunto de permisos asignados a los roles que están activados en todas las sesiones del usuario, sin tener en cuenta las sesiones establecidas por otros usuarios en el sistema. RBAC añade la posibilidad de modelar una jerarquía de roles de forma que se puedan realizar generalizaciones y especializaciones en los controles de acceso y se facilite la modelización de la seguridad en sistemas complejos

2.2.4. Modelo TBAC

El control de acceso basado en tareas (TBAC, Task Based Access Control) permite controlar el acceso en entornos representados por workflow. El modelo TBAC extiende los tradicionales modelos de control basados en sujetos/objetos incluyendo aspectos que aportan información contextual basada en las actividades o tareas [24]. El control de acceso en TBAC es garantizado por medio de “Etapas de autorización”. Las “Etapas de autorización” son un concepto abstracto introducido por TBAC para modelar y manejar un sistema de permisos relacionados con el progreso de las tareas o actividades dentro del contexto de un workflow. Este concepto esta compuesto por una serie de elementos y atributos. A continuación se describen los elementos más representativos:

- Estado del Proceso: Indica como ha progresado la “etapa de autorización” en su ciclo de vida.

- Estado de Protección: Define todos los permisos que pueden ser activados por la “etapa de autorización” y que son mantenidos por la propia “etapa de autorización”. El valor del estado de protección, en un momento dado, nos da una instantánea de los permisos activos en ese momento. El contenido del estado de protección puede cambiar en base al proceso de la tarea o a la perdida de validez de los permisos. Esto ultimo es debido a que con cada permiso se asocia una especificación de validez y de uso que nos detalla las condiciones que hay que cumplir para que los permisos asociados con una “etapa de autorización” se han válidos y puedan ser usados. El estado de protección de cada “etapa de autorización” es único y disjunto con respecto a los estados de protección de otras etapas.

- Conjunto de administradores: Contiene información relevante acerca del conjunto de administradores que potencialmente pueden conceder/invocar la “etapa de autorización” así como sus identidades de usuario y sus roles.

- Administrador Ejecutor: Identifica el miembro del conjunto de administradores que eventualmente invoca la “etapa de autorización”.

2.3. Vulnerabilidades Riesgo y Seguridad

En la ingeniería de software el paradigma C.I.A. pertenece al dominio de los requerimientos, estableciendo los objetivos de mas alto nivel relacionados con la seguridad de la información. La arquitectura A. A. A. y sus componentes son especificaciones de software y hardware de la arquitectura de sistemas en cual se esfuerza para implementar esos requerimientos. Por lo tanto los sistemas de seguridad son las implementaciones practicas de esas especificaciones. La confianza puesta en ese proceso puede ser expresado en términos de “Garantía” (assurance) [25] La garantía puede ser definida como la base para las medidas de “Seguridad”, tanto los trabajos operacionales como técnicos orientados a proteger los sistemas, los procesos de información y los objetivos de seguridad, integridad, disponibilidad y confidencialidad han logrado

encontrar una implementación especifica. En un ambiente ideal se puede responder en forma perfecta a las especificaciones y las especificaciones pueden cumplir y exceder los requerimientos. Sin embargo es evidente de que el ambiente no es ideal y por lo tanto hay debilidades que afectan el camino entre requerimientos y aplicaciones. Estas debilidades se pueden resumir en las siguientes:

- Debilidades de Análisis al establecer los requerimientos de confidencialidad, integridad y disponibilidad para los activos de la información.
- Debilidades de diseño mientras se trasladan los requerimientos de alto nivel en especificaciones en términos de políticas y arquitecturas para autenticación, autorización y auditoría. Debilidades de implementación mientras se codifica, implementa y configura los sistemas de seguridad.

Adicionalmente los requerimientos de seguridad no son estables sino que interactúan en forma continua con el medio y por lo tanto es necesario un ciclo de desarrollo de para mantener la seguridad de los sistemas en forma permanente adaptándose a las cambiantes necesidades de los mercados.

2.4. Distinción entre Amenaza, Vulnerabilidad y Riesgo

La seguridad de la información es por si solo la ciencia de lo incierto. La seguridad de la información se debe manejar de acuerdo a la administración de riesgos que se esta dispuesto a correr, la seguridad absoluta no existe o es infinitamente cara, por lo tanto se debe medir el riesgo de perdida o afectación de la seguridad de la información para determinar la inversión en seguridad a realizar.
Varias normas ISO [3] [26] definen claramente las diferencias entre Amenaza, Vulnerabilidad y Riesgo:

Riesgo Combinación de la probabilidad de un evento y su consecuencia.

Amenaza Causa potencial de un incidente no deseado, el cual va resultar en un daño a los sistemas u organización.

Vulnerabilidad Debilidad de un activo o grupo de activos que pueden ser explotados por una o más amenazas.

Las tareas relacionadas con administrar y reducir los riesgos relacionados con el uso de la información, para reducir o manejar las vulnerabilidades o amenazas. Es un error pensar la seguridad en términos de reducir las vulnerabilidades. La seguridad es un componente del proceso de la administración de riesgos de la organización Dicho de otra forma la seguridad de la información es la protección de la información de un amplio rango de amenazas en orden a asegurar la continuidad de las organizaciones, minimizar los riesgos y maximizar el retorno de las inversiones y las oportunidades de negocios.
Los componentes de la administración de riesgos son:

- Análisis de riesgos: uso sistemático de la información para identificar las fuentes de riesgo y estimarlos.
- Evaluación de riesgos: El proceso de comparar el riesgo estimado contra el riesgo el criterio de riesgo dado para determinar el significad del riesgo.
- Auditoría de riesgos: Todo el proceso de análisis de riesgo y evaluación de riesgo
- Tratamiento de los riesgos: El proceso de selección e implementación de las medidas para reducir los riesgos.

2.5. Ataques a Computadoras

Desde el momento en que se comenzó a interconectar computadoras para formar redes, aparecen las amenazas, Anderson[1], en su temprano trabajo en el año 1972 expone el concepto de acción maliciosa en los servidores y los intentos de penetración.
En el año 1983 el DoD (Department of Defence) publica una de las primeras y más difundidas

normas de seguridad, y la actualiza el año 1985 [2].

En el 2000 la Internet Society network working group elabora un glosario de seguridad informática [6] y la ITU a través de la norma .X800 [4], en las cuales se hace referencia a los principales tipos de ataques a las redes de computadoras, distinguiendo entre ataques pasivos y activos, por otro lado Bruce Schneier [27] distingue siete tipos de ataques sin realizar ninguna distinción, entre pasivos o activos, se ha preferido utilizar la clasificación de la RFC2828 que se expone a continuación:

- Ataques Pasivos
- Ataques Activos

2.6. Ataques Pasivos

Estos son los de escucha sin autorización o de monitoreo de tráfico. Los objetivos de estos ataques consisten en obtener la mayor cantidad de información del mensaje transmitido y del oponente. Las distintas modalidades de ataques pasivos son las siguientes:

- Descarga de contenidos del mensaje: Están incluidos dentro de este tipo de ataque la escucha de una conversación telefónica, la lectura de un mensaje de correo electrónico o la información confidencial capturada por un oponente.
- Análisis de tráfico: Este es un ataque muy sutil. Se supone que hay medios de envíos de mensajes confidenciales, que no permiten al atacante poder acceder al contenido del mensaje. El atacante tiene sólo la posibilidad de observar la transmisión de los mensajes y obtener de éstos, por ejemplo datos tales como: la frecuencia de emisión de los mensajes y la longitud del mensaje. Esta información puede ser de mucha ayuda para inferir la naturaleza de la comunicación.

Los ataques pasivos son muy difíciles de detectar y reconocer, porque ellos son un medio de reconocimiento previo a la realización de ataques activos.

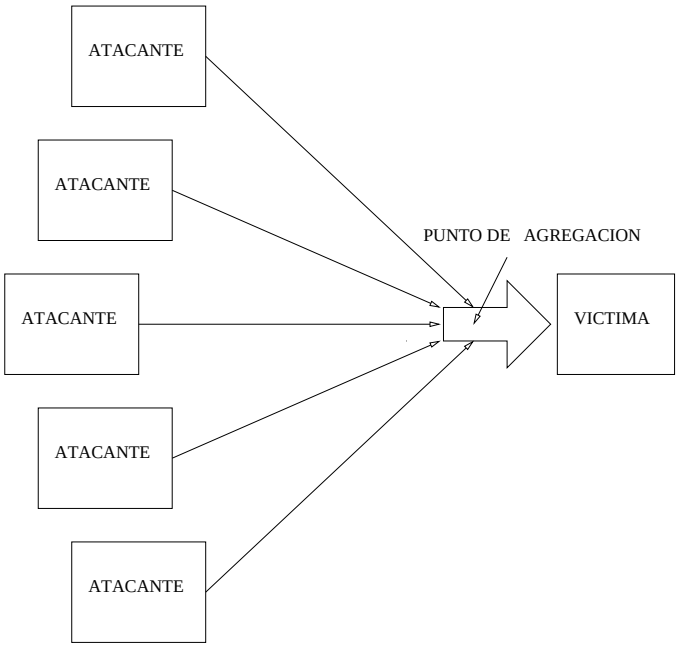

Figura 2.1: Inundación distribuida de una red.

2.7. Ataques Activos

Los ataques activos involucran y comprometen los pilares básicos de las prácticas de seguridad: la confidencialidad, la integridad y la disponibilidad (CIA Confidentiality, Integrity and Availability)[28]. Los ataques activos son:

- Denegación de Servicio DoS (Denegation od Service): El efecto de este ataque es impedir la posibilidad de acceso a toda persona a un determinado servidor.
- "Masquerade" (Enmascarado): En este caso el atacante se representa él mismo como un legítimo usuario con el objeto de robar, alterar o destruir recursos informáticos.
- "Replay" (Reinterpretar): Este ataque es llevado a cabo mediante una captura pasiva de datos, para que luego sean retransmitidos y con ello producir efectos no autorizados.
- Modificación de contenidos del mensaje: La información original es alterada de tal forma que permita obtener un resultado no autorizado.

Al tipo de ataque denegación de servicio se desarrollará más en detalle debido a que este trabajo se focaliza en este tipo de ataque. Existen básicamente dos tipos de ataques de denegación

de servicio según el ataque provenga de una fuente DoS (Denegation of Service) o de varias fuentes Denegación de servicio distribuida DDoS (Distribute Denegation of Service). ver fig. 2.1

Uno de los ataques más comunes de denegación de servicio, se produce cuando se establece una conexión Internet, con el protocolo de Transporte de Flujo Confiable (TCP Transport Control Protocol) desde un cliente a un servidor y el cliente envía un paquete de sincronización (SYN), el servidor responde con un paquete de reconocimiento de sincronización (SYN ACK), esperando el reconocimiento del cliente (ACK), para esta operación el servidor crea una cola de tamaño finito esperando que la conexión se complete, si el atacante envía una cantidad suficientemente grande de solicitudes de conexión sin completarlas, produce un desborde de la cola, este tipo de ataque se conoce como inundación TCP- SYN.

2.7.1. Ataques de Inundación

2.7.1.1. Síntomas de los ataques

El CERT (Computer Emergency Readiness Team) de Estados Unidos,[29] expone los síntomas de los ataques de denegación de servicio los cuales incluyen las siguientes manifestaciones:

- Lento rendimiento de la red (apertura de los archivos o el acceso a sitios web)
- Falta de disponibilidad de un sitio web en particular.
- Imposibilidad a acceso a cualquier sitio web.
- Aumento dramático en el número de “spam” recibidos mensajes de correo electrónico - (este tipo de ataque DoS es llamado “Bomba de Mail”.)

No todas las interrupciones de los servicios, incluso aquellos que son el resultado de la actividad maliciosa, son necesariamente de ataques de denegación de servicio. Otros métodos de ataque, pueden incluir una denegación de servicio como un componente de una mayor ofensiva. Ataques de denegación de servicio puede también dar lugar a problemas en la red alrededor de la computadora bajo ataque. Por ejemplo, el ancho de banda de un “router” entre Internet y

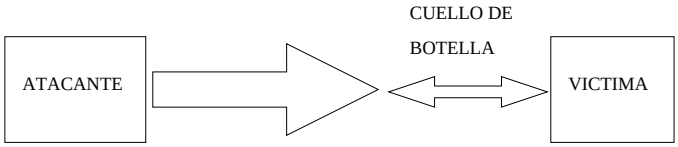

Figura 2.2: Inundación de una red.

la red local pueden ser consumidos por un ataque DoS, poniendo en peligro no sólo el equipo, sino también toda la red. Si la denegación de servicio se realiza a una escala suficientemente grande, toda la región geográfica de la conectividad a Internet puede verse comprometida, sin conocimiento del atacante o por una mala configuración de la infraestructura de la red.

2.7.1.2. Métodos de ataque

El ataque de "denegación de servicio" se caracteriza por un explícito intento de los atacantes para evitar que los usuarios legítimos de un servicio realicen uso de este. Los ejemplos incluyen:

- Inundación de una red, evitando así el tráfico de red legítimo ver fig. 2.2;
- Interrumpir un servidor mediante el envío de más solicitudes de lo que posiblemente puede manejar, lo que impide el acceso a un servicio;
- Impedir a una persona en particular el acceso a un servicio;
- Interrumpir el acceso a un servicio específico a una persona.

Los ataques se pueden enviar a cualquier dispositivo de red, incluidos los ataques a los dispositivos de enrutamiento y acceso a la Web, correo electrónico, o al servicio de Sistemas de Nombre de Dominio DNS (Domain Name System).

Un ataque de denegación de servicio pueden ser perpetrados de diferentes formas. Existen tres tipos básicos de ataques de acuerdo al CERT [30]:

1. El consumo de recursos computacionales.
    - Conectividad de la Red.
    - Uso de los propios recursos en contra de uno mismo.

- Ancho de banda.
- Consumo de otros recursos como: Espacio en disco; Tiempo de CPU.

2. Perturbación de la información de configuración, como la información de enrutamiento;información de estado, peticiones no solicitada de reiniciar sesiones TCP;
3. Perturbación física de los componentes de la red.

Un ataque de denegación de servicio puede incluir la ejecución de malware destinados a:

- Maximizar el uso de la CPU evitando la ejecución de cualquier tipo de trabajo;
- Desencadenar errores en el microcódigo de la máquina;
- Desencadenar errores en la secuencia de instrucciones, con el fin de forzar a la computadora a un estado de inestabilidad o de inmovilización;
- Explota errores en el sistema operativo de recursos a causa del agotamiento de estos obligándolo a a utilizar todos los recursos disponibles por lo que no no se puede ejecutar el trabajo real;
- Caída sistema operativo;
- IFrame DDoS, en la que un documento HTML hace requerir una página web con muchos KB de información, hasta que alcanzar la cantidad de visitas que da lugar a que se supere el límite de ancho de banda.

2.7.2. Principales Ataques DoS

A continuación se describen los principales ataques de denegación de servicio.

2.7.2.1. Ataque Smurf

Un ataque smurf es una variante particular de un ataque de denegación de servicio de inundación en Internet [31][32]. Se basa en la configuración erróneas de los dispositivos de red que

permiten a los paquetes que se envían a la red a través de la dirección de difusión, en lugar de a una máquina específica. Normalmente se utilizan paquetes ICMP de solicitud de eco. La red sirve entonces como un “smurf” amplificador. En ese ataque, los atacantes envían un gran número de paquetes IP con la dirección de la fuente falsa, en la dirección de la fuente se coloca la dirección de la víctima. Para luchar contra ataques de Denegación de Servicio en Internet[33], tal como el ataque (Smurf) Amplificador, los proveedores de servicio de Internet (ISP) cuentan con la capacidad de identificar configuraciones erróneas de las redes y de adoptar medidas correctivas como el filtrado de paquetes por direcciones de origen.

2.7.2.2. Inundaciones Ping

Inundaciones Ping se basa en el envío a la víctima un número muy grande de paquetes ping, por lo general a través del comando "ping -f". Es un ataque muy sencillo de realizar, la exigencia principal es que el atacante tenga acceso a un ancho de banda mayor que la de la víctima.

2.7.2.3. Inundación TCP-SYN

Los ataques más comunes de denegación de servicio, se produce cuando se establece una conexión Internet, con el protocolo de Transporte de Flujo Confiable (TCP Transport Control Protocol) desde un cliente a un servidor y el cliente envía un paquete de sincronización (SYN), el servidor responde con un paquete de reconocimiento de sincronización (SYN ACK), esperando el reconocimiento del cliente (ACK), para esta operación el servidor crea Block de Control TCP (TCB TCP Control Block) de tamaño finito, puede ocupar de 280 Bytes hasta 1300 Bytes dependiendo del sistema operativo [34], esperando que la conexión se complete, si el atacante envía una cantidad suficientemente grande de solicitudes de conexión sin completarlas, produce un desborde de la cola de TCB abiertos, este tipo de ataque se conoce como inundación TCP- SYN. Este tipo de ataque los veremos mas en detalle debido a que es uno de los ataque que nos proponemos trabajar para mitigar su efecto.

2.7.2.4. Ataque Teardrop

El ataque consiste en el envío de paquetes IP fragmentados de tal forma que los fragmentos se superpongan, provocando sobrecargas en la computadora de destino. Los elementos manejados son la superposición de fragmentos, más el tamaño grande de estos provocan sobrecarga en la computadora de destino.
Un fallo en el protocolo de fragmentación y rearmado de los paquetes TCP/IP de diversos sistemas operativos causa que los fragmentos no estén bien manipulados, y falle el rearmado, los sistemas operativos Windows 3.1x, Windows 95 y Windows NT, como en Versiones de Linux anteriores a 2.0.32 y 2.1.63 son vulnerables a este ataque [35].

2.7.2.5. Ataques Peer-to-peer

Los atacantes han encontrado una forma de explotar una serie de errores en los servidores peer-to-peer para iniciar ataques DDoS. El más agresivo de estos ataques DDoS peer-to-peer, explota DC ++ (Es un cliente "open source" para windows para compartir archivos). Los ataques Peer-to-peer son diferentes de los ataques basados en botnet (botnet es un termino usado para designar una colección de programas robots "bots" los cuales pueden ser ejecutados de manera autónoma y en forma automática. Ellos se ejecutan en computadoras zombie controladas en forma remota). [36]. Con el ataque peer-to-peer no hay botnet y el atacante no tiene que comunicarse con los clientes que desea atacar. En lugar de ello, el atacante actúa como un "titiritero", instruyendo a los clientes peer-to-peer para desconectarse de su par y conectarse a la víctima. Como resultado de ello, varios miles de computadoras pueden agresivamente tratar de conectarse a una página web. Si bien un típico servidor web puede manejar unos cientos de conexiones por segundo antes de comenzar a degradar el servicio, la mayoría colapsa instantáneamente con cinco o seis mil conexiones por seg. Un ataque moderado podría generar un máximo de 750000 conexiones en un corto lapso de tiempo.

El servidor web bajo ataque recibirá conexiones entrantes confusas Aunque los ataque peer to peer son fáciles de identificar con firmas, el gran número de direcciones IP que se deben bloquear (a menudo más de 250000 en el curso de un gran ataque) significa que este tipo

de ataque puede sobrepasar las defensas del firewall Incluso si un firewall pueden mantener el bloqueo de direcciones IP, existen otros problemas a considerar. Por ejemplo, hay un breve momento en que la conexión se abre en el lado del servidor antes de que la firma del ataque llega a identificarse. Sólo una vez que se abre la conexión con el servidor puede ser identificada la firma y bloqueada la conexión. Incluso el bloqueo de las conexiones puede agotar los recursos del servidor

2.7.2.6. Inundaciones a nivel de Aplicación

Inundaciones IRC constituye un ataque común a nivel de aplicación.

Varios DoS exploits causan desbordamiento de buffer que pueden provocar que el software que se esta ejecutando en el servidor llene el espacio en el disco o consuma toda la memoria o tiempo de CPU [37].

Otros tipos de denegación de servicio se basan principalmente en la fuerza bruta, las inundaciones con un abrumador flujo de paquetes, saturando el ancho de banda de la conexión a Internet o agotando los recursos del sistema. Las inundaciones por saturación del ancho de banda dependen de que el atacante tenga mayor ancho de banda disponible que la víctima, una forma de lograr esto hoy es a través de denegación de servicio, distribuida utilizando un botnet. Otras inundaciones pueden utilizar tipos específicos de paquetes o las solicitudes de conexión para saturar los limitados recursos de la víctima, por ejemplo, ocupando el número máximo de conexiones abiertas o llenando el espacio de registro de la víctima.

2.7.2.7. Ataque banana

Se trata de reorientar los mensajes enviados desde el cliente de vuelta hacia el cliente, evitando la entrada de paquetes de afuera de la red, inundando al cliente con sus propios paquetes[38].

Un atacante con acceso a un equipo de la víctima puede disminuir la velocidad de este hasta que esta sea inusable o usando una bomba de procesos.

2.7.2.8. Ataque zombie

"Ataque Zombie" es un término que hace referencia a un ataque de denegación de servicio. Una red es objeto de hostilidad por diferentes atacantes haciendo ping a las computadoras durante un largo período de tiempo. El resultado es una degradación de la calidad de servicio y un incremento de la carga de trabajo para los recursos y la red. Este tipo de ataque es más difícil de detectar que los tradicionales ataques de denegación de servicio debido a su naturaleza encubierta.

2.7.2.9. Ataque Nuke

Nuke es un viejo ataque de denegación de servicio contra las redes que consiste en el envío de paquetes ICMP fragmentados o paquetes ICMP inválidos esto se logra mediante una modificación a la utilidad ping que provoca el envió repetido de datos corruptos, provocando ralentizar la computadora afectada, hasta que llega a un alto total.

En los juegos de azar en línea, es utilizado por vikingo difundiendo a otro usuario, o a todos los demás usuarios, con mensajes repetidos al azar en rápida sucesión. Dichas técnicas también se observan en los programas de mensajería instantánea, como en repetidas ocasiones el envío de texto se pueden asignar a una macro o AppleScript. Los sistemas operativos modernos son generalmente resistentes a estos ataques nuke, y ahora tienen los juegos en línea de terceras partes "Control de inundaciones".

Un ejemplo concreto de un ataque nuke que adquirido cierta prominencia es la WinNuke, que explotan la vulnerabilidad en el manejador de NetBIOS en Windows 95. Una cadena de fuera de la banda de datos se envió al puerto TCP 139 de la máquina de la víctima, causando que se cuelgue y mostrar una Pantalla Azul de la Muerte.

2.7.2.10. Ataques distribuidos

Un ataque distribuido de denegación de servicio (DDoS) se produce cuando varios sistemas generan una inundación comprometiendo el ancho de banda o recursos de un sistema, por lo

general uno o más servidores web. Estos sistemas están comprometidos por atacantes usando una variedad de métodos.

Un Malware puede transportar mecanismos de ataques DDoS; uno de los más bien conocidos ejemplos de esto fue MyDoom. Su mecanismo de ataque de DoS fue activado en una fecha y hora específicas. Este tipo de ataque DDoS tiene pregrabada la dirección IP de la víctima y no requiere mayor interacción una vez lanzado el ataque.

Un sistema también pueden ser comprometido con un troyano, lo que permite al atacante descargar un agente zombie (o el troyano puede contener uno). Los atacantes también pueden introducirse en los sistemas utilizando herramientas automatizadas que explotan las fallas en los programas que están a la escucha de las conexiones desde Host remotos. Este escenario se refiere principalmente a los sistemas que actúan como servidores de la web.

Stacheldraht es un clásico ejemplo de una herramienta de DDoS. Utiliza una estructura de capas donde el atacante utiliza un programa cliente para conectarse a los manipuladores, que están en el sistema comprometido enviando los comandos al agente zombie, que a su vez facilita el ataque DDoS. Los agentes se comprometen a través de los manipuladores por el atacante, utilizando rutinas automatizadas para explotar las vulnerabilidades en los programas que aceptan conexiones remotas. Cada manejador puede controlar hasta a un millar de agentes.

Estas colecciones de sistemas comprometidos son conocidos como botnets. Herramientas DDoS como Stacheldraht todavía utilizan métodos de ataque DoS clásicos centrado alrededor de IP spoofing y amplificación como smurf, estos ataques también se conocen como ataques de consumo de ancho de banda. Inundaciones SYN (también conocidos como ataques de consumo de recursos) también pueden ser utilizada. Nuevas herramientas pueden utilizar servidores DNS para fines de DoS. A diferencia de MyDoom los mecanismo DDoS, botnets se pueden dirigir contra cualquier dirección IP. Script kiddies usan para negar la disponibilidad de los sitios web bien conocidos a los usuarios legítimos. Más sofisticadas herramientas DDoS los atacantes hacen uso con fines de extorsión - incluso en contra de sus rivales de negocios.

Es importante señalar la diferencia entre un ataque DoS y DDoS. Si un atacante monta un ataque smurf por parte de un único Host que sería clasificado como un ataque DoS. De hecho,

cualquier ataque contra la disponibilidad sería clasificado como un ataque de Denegación de Servicio. Por otra parte, si un atacante utiliza mil zombie a la vez los sistemas de lanzamiento de ataques smurf contra un Host remoto, éste sería clasificado como un ataque DDoS. Las principales ventajas para un atacante de la utilización de un ataque de denegación de servicio distribuido es que múltiples máquinas que pueden generar más tráfico que un ataque de la máquina, múltiples máquinas de ataque son más difíciles de apagar que el ataque de una sola máquina, y que el comportamiento de cada máquina de ataque Puede ocultarse mejor, lo que lo hace más difícil de detectar y evitar. Estas ventajas del atacante pueden causar problemas en los mecanismos de defensa. Por ejemplo, si se limita a la compra de más ancho de banda de entrada que el actual volumen del ataque, podría no ser una solución valida, ya que el atacante podría simplemente añadir más máquinas de ataque.

2.7.2.11. Ataque reflejado

Un ataque de denegación de servicio distribuido reflejado(DRDoS) incluye el envío de solicitudes de algún tipo a un gran número de ordenadores que responderán a las peticiones. Usando el protocolo Internet TCP y realizando spoofing de la dirección de origen de tal forma que sea la de la víctima, todas las repuestas estarán dirigidas hacia la víctima provocando la inundación.

El ataque de solicitud de eco ICMP (Smurf Attack) puede considerarse una forma de ataque reflejado. El servidor atacante solicita ecos ICMP a la dirección de broadcast de una red mal configurándola realizando spoofing de la dirección de origen, provocando que muchos servidores respondan provocando una inundación. Algunos de los primeros programas de denegación de servicio distribuida aplicaban los métodos de este ataque.

Muchos servicios pueden ser explotadas para actuar como reflectores, algunos más difíciles de bloquear que otros. Ataque de amplificación DNS un nuevo mecanismo que aumenta el efecto de amplificación, utilizando una lista mucho más amplia de los servidores DNS que ya se ha visto.

2.7.2.12. Ataques no intencionados

Ataques no intencionados se describen como una situación en la que en un sitio web se produce una denegación de servicio, no debido a un ataque deliberado por un solo individuo o grupo de individuos, sino simplemente debido a un súbito repunte en la popularidad. enorme. Esto puede suceder cuando un sitio web muy popular pone enlace a un segundo sitio web, menos preparado, para recibir un gran número de peticiones, por ejemplo, como parte de una noticia. El resultado es que una proporción significativa de los principales usuarios del sitio ordinario - potencialmente cientos de miles de personas - haga clic en este vínculo en el espacio de unas horas, que tienen el mismo efecto en la página web como un objetivo de ataque DDoS.

Sitios de noticias y los sitios de paginas de enlaces, sitios cuya función principal es proporcionar enlaces a otros lugares interesantes de contenido en Internet, son más susceptibles de causar este fenómeno. El ejemplo canónico es el efecto Slashdot. Sitios como Digg, Fark, Algo Malo, y el webcomic Penny Arcade tienen sus correspondientes efectos, conocido como "el efecto Digg", "farking", "goonrushing" y "wanging"; respectivamente.

Routers también se han conocido que pueden crear ataques de denegación de servicio no intencional, como D-Link y Netgear, Routers han creado "NTP vandalismo" por las inundaciones a servidores NTP sin respetar las restricciones de los tipos de clientes o limitaciones geográficas.

Ataques involuntarios similares también pueden ocurrir a través de otros medios, por ejemplo, cuando se menciona una dirección URL en la televisión. Si un servidor está siendo indexado por Google u otro motor de búsqueda durante los períodos de máxima actividad, o no tiene una gran cantidad de ancho de banda disponible mientras transcurre la indexación, también pueden experimentar los efectos de un ataque DoS.

2.8. Clasificación basada en valorización de riesgos

Nong Ye [39] propone una clasificación de ataques basada en la teoría de sistemas en la valorización de riesgos, la cual es muy útil en el momento de de evaluar las estrategias de

Cuadro 2.1: Clasificación basada en valorización de riesgos Causas.

Causa					
Objetivo	Propagación	Origen	Acción	Vulnerabilidad	Activo
Espionaje	Humana	Local	Prueba	Configuración	Red
Crimen	Autónoma	Remoto	Búsqueda	Diseño	Servidor
Terrorismo			Inundación	Implementación	Proceso
Rivalidad			Autenticar		Datos
Cracking			Puente		Usuario
Vandalismo			Engaño		
			Leer		
			Copiar		
			Terminar		
			Borrar		

mitigación de ataques basadas en el riesgo que representa para los sistemas. Para realizar una clasificación con utilidad práctica en la detección de anomalías es importante el trabajo de Howard [40] en el cual establece una taxonomía para el lenguaje usado en los incidentes de seguridad.

De acuerdo a la teoría de modelado de fallas una primera clasificación es separar las causas de los efectos y finalmente basado en la teoría de la ingeniería en sistemas, los efectos pueden ser sobre el estado o sobre el rendimiento de los sistemas.

En el cuadro 2.1 se presenta las causas de acuerdo a la subclasificación de Objetivos, Propagación, Origen, Acción, Vulnerabilidad y Activos que afecta.

En el cuadro 2.2 se presenta la subclasificación de efectos de acuerdo a que estos afecten el estado o el rendimiento de los activos a proteger.

De acuerdo a Howard [40] es difícil realizar una clasificación de ataques que cumpla los requisitos de exclusión mutua, exhaustiva, no ambigua, repetible, aceptable y usable, siendo el primer principio el más difícil de cumplir basándose en la experiencia de clasificación del

Cuadro 2.2: Clasificación basada en valorización de riesgos Efectos.

Efecto	
Estado	Rendimiento
Disponibilidad	Tiempo de ejecución
Integridad	Precisión
Confidencialidad	Exactitud
Ninguna Acción	Ninguna Acción

CERT [41] y una lista exhaustiva seria demasiado larga para que se practica.

Capítulo 3

IDS

3.1. Sistema de detección de intrusiones

La detección de intrusión en redes IDS es un componente vital en la defensa contra ataques a redes y es abordado desde diferentes perspectivas como puede verse en el trabajo de Bai [42]. En él se implementan dos métodos principales de detección: La detección de intrusiones de redes NID (Network Intrusion Detection), y la detección de intrusiones de servidores HID (Host Intrusion Detection).

3.1.1. Detección de intrusiones de redes

Los sistemas NID están relacionados con el tráfico de información entre servidores y clientes Típicamente referidos como espías de paquetes (packet-sniffers), estos dispositivos interceptan paquetes que viajan por los medios de comunicación y transportan datos encapsulados en diferentes protocolos tales como, “Frame Relay” o enlaces en Modo de Transferencia Asincrónica (ATM Asynchronous Transfer Mode). Algunos dispositivos NID comparan el paquete con una base de datos de firmas de ataques conocidos y huellas digitales de paquetes maliciosos, mientras que otros analizan la actividad de paquetes buscando un comportamiento anómalo que pueda ser malicioso. En cualquiera de ambos casos un dispositivo IDS debe ser visto principalmente como una defensa perimetral de la red.

Los dispositivos NID en el pasado y por la complejidad de la tarea que realizan, han sido incapaces de operar en los siguiente ambientes:

- Redes conmutadas
- Redes encriptadas
- Redes de alta velocidad

Recientemente esta limitación ha sido superada por la potencia de procesamiento y cálculo de los microprocesadores modernos. Los conmutadores de redes ya vienen equipados con dispositivos IDS, capaces inclusive de realizar "packet-sniffing" (espías de paquetes) a velocidades de giga bit por segundo. La técnicas de NIDS pueden desagregarse en dos tendencias principales y complementarias entre sí: Detección de mal uso y detección de anomalías. La detección de mal uso como lo explica Vigna [43] modela ataques conocidos y realiza búsquedas de la ocurrencia de esos patrones. Los sistemas de detección de anomalías, como lo señala Valdes [44] alertan de intrusiones mediante la observación de las desviaciones del comportamiento típico del tráfico de la red.

3.1.2. Detección de intrusiones de servidores

La detección de intrusiones basada en servidores HIDS (Host Intrusion Detection Systems), está diseñada para responder a ataques sobre un determinado servidor. Se basan en la supervisión de las acciones de los usuarios y de los archivos del servidor. En auditoría de los registros de actividad de los servidores y estado de los archivos del sistema, existen técnicas robustas que ofrecen administración de políticas de auditoría, análisis estadístico y soporte de evidencias, los que proveen medidas de control de la actividad de los servidores. La detección de intrusión en servidores sirve tanto para detectar ataques externos como internos.
Dentro de los HIDS se considera una división especial a los sistemas detectores de Archivos los cuales mediante una firma detectan cualquier cambio en el sistema de archivos.

3.1.3. IDS Centralizados vs Distribuidos

De acuerdo a los mecanismos de control un sistema IDS puede ser clasificado como centralizado o distribuido. Un sistema centralizado es aquel en que las tareas de monitoreo y control son realizadas desde un lugar fijo y central tal como el descripto en [45]. La ventaja de un mecanismo centralizado es la facilidad para desarrollarlo y mantenerlo, pero significa un cuello de botella cuando el tráfico en la red es grande y además se convierte en un único punto de falla. En un sistema distribuido el mecanismo de detección y análisis se puede realizar por diversos agentes cooperantes y autónomos a los cuales se les asigna tareas especificas de detección [46]. La principal ventaja de sistemas distribuidos es que los agentes autónomos trabajan en paralelo y pueden reaccionar en forma más rápida a tráfico mutable de la red.

3.1.4. Sistemas basados en Firmas

Gran parte de los IDS trabajan buscando patrones conocidos dentro del tráfico TCP/IP, mediante técnicas conocidas como inspección profunda de paquetes (DPI), estos métodos tienen sus ventajas y sus desventajas dentro de las cuales podemos citar:

- Son simples: comparan el tráfico de la red con firmas conocidas de ataques. Si una porción del tráfico coincide con un ataque conocido, el IDS genera una alarmas y almacena un alerta en el registro.
- Necesitan un bajo mantenimiento: Solo se necesita mantener una base de datos de firmas actualizadas.
- Tienden a registrar un porcentaje muy bajo de falsos positivos: Atributo altamente apreciado entre los administradores de redes. Un porcentaje alto de falsos positivos sobrecarga de trabajo innecesario a los administradores.
- Tienen una limitación importante: no son efectivos contra nuevos ataques y variaciones de los ataques conocidos.

3.1.5. Sistemas basados en Detección de anomalías

Los sistemas de detección de anomalías se basan en procesos estadísticos, redes neuronales y agentes inteligentes autónomos. Las principales características de estos son:

- Tienden a ser complejos: determinar que constituye un funcionamiento normal del tráfico de una red es una tarea para nada trivial, por lo que suele usarse algunos de los métodos citados anteriormente.
- Necesitan mucho mantenimiento: normalmente requieren un largo periodo de inicialización o entrenamiento durante el cual se recoge información necesaria para su buen funcionamiento.
- La detección de anomalías suele tener muchos falsos positivos.

3.2. Productos IDS

Existe una gran variedad de productos IDS tanto comerciales como de dominio público [47].
A los productos actualmente en vigencia se trataran en párrafo aparte. A continuación se enumeran algunos proyectos importantes que realizaron un significativo aporte al desarrollo de los IDS.

- Sistema de defensa automático, escalable y flexible (An Automated, Dynamic, Flexible, Distributed, Scalable Network Defense) Año 1998 Autores Fred Cohen, Eli Dart, Tim Berg, Cindy Phillips, Vitus Leung, y Stefan Chakerian, este programa fue patrocinado por el U.S. Department of Energy - Defense Programs Organization y Sandia National Laboratories a partir de 1998 no se han tenido más noticias de este desarrollo.
- Sistema de detección de intrusiones para ruteo externo (External Routing Intrusion Detection System ERIDS) Contrato Número: F30602-98-c-0242, desarrollado por BBN Technologies Internetwork Research Department. Autores: Stephen T. Kent, Luis A. San-

chez. Objetivos: Desarrollar un sistema de detección de intrusiones ERIDS capaz de proveer a los operadores de los Centros de Operación de Redes (NOCs), con información indicativa del funcionamiento incorrecto del ruteo externo BGPv4 debido a ataques maliciosos o mala configuración de estos año 1999.

- Monitoreo de eventos generando repuestas en tiempo real a anomalías (Event Monitoring Enabling Responses to Anomalous Live Disturbances EMERALD), fue desarrollado por Phillip Porras EMERALD representa el estado del arte en 2001 en investigación y desarrollo de componentes y sistemas para la detección de anomalías y mal uso en redes [48], entre los principales objetivos figuran:

  - Vigilancia de redes escalable.
  - Alto volumen de análisis de eventos.
  - Sensores livianos distribuidos.
  - Infraestructura genérica y compartimentada.
  - Fácil de adaptar para redes nuevas y políticas especificas.

  Este desarrollo fue financiado por Defense Advanced Research Projects Agency (DARPA).

  EMERALD usa sistemas expertos en inferencia basadas en reglas PBEST [49], también incluye componentes basado en inferencia adaptiva Bayesiana [50].

- Sentivist IPS. Identifica y bloquea ataques maliciosos a la red, antes que puedan causar daño a la organización. Trabaja decodificando todo el protocolo de la capa de aplicación no solo los protocolos comunes tales como HTTP, SMTP, DNS, FTP, SMB, MSRPC, etc, sino también protocolos menos usados como SIP (VoIP) y algunos P2P y protocolos de mensajería instantánea. Además de la detección a nivel de aplicación, Sentivist IPS detecta y previene ataques tales como Inundaciones, ataque de fuerza bruta, Troyanos. Gusanos, también tiene las funcionalidades de un firewalll. El 8 de enero de 2007 Check Point Software Technologies Ltd compro a Sentivist IPS.

3.2.1. Productos IDS Comerciales

Los productos comerciales los podemos agrupar en tres sub-categorías de acuerdo al alcance de la protección ofrecida.

- Detección de Intrusiones (Intrusion Detection Systems IDS). No existen productos comerciales que se limiten a solo detectar las intrusiones sino que que ofrecen una solución integral.

- Prevención de intrusiones (Intrusion Prevention System IPS) La mayoría de los productos comerciales se agrupa bajo esta denominación, detectando y previniendo intrusiones.

- Administradores de Incidentes unificados (Unified threat management UTM). Los nuevos productos del mercado se orientan a un tratamiento unificado de la seguridad proveyendo una solución para cualquier tipo de incidente y comprende típicamente los siguientes módulos:

    - Firewall
    - VPN
    - IDS/IPS
    - Anti Virus
    - Anti Spam
    - Filtrado URL
    - Filtrado de contenidos.

Otra clasificación de los productos IDS (IPS) es de acuerdo a la licencia de uso, productos de Licencia Propietaria y Licencia Open Source.

Existe una gran variedad de productos IPS de licencia propietaria, la búsqueda de productos no es exhaustiva, sin embargo los sitios de internet que contienen evaluaciones de productos IPS incluyen a los aquí considerados.

- IBM http://www.iss.net
- Cisco http://www.cisco.com
- McAfee http://www.mcafee.com
- Secureworks http://www.secureworks.com
- Juniper http://www.juniper.net
- ChekPoint http://www.checkpoint.com
- Radware http://www.radware.com
- Deep Nine http://www.dipnines.com
- Enterasys http://www.enterasys.com
- Securesoft http://www.securesoft.co.jp
- Toplayer http://www.toplayer.com
- Nitrosecurity http://www.nitrosecurity.com
- Broadweb http://www.broadweb.com
- Fortinet http://www.fortinet.com
- Sourcefire http://www.sourcefire.com
- Stonesoft http://www.stonesoft.com
- Tipping Point http://www.tippingpoint.com
- Reflex Security http://www.reflexsecurity.com
- Still secure http://www.stillsecure.com
- TrustWave https://www.trustwave.com

- NETASQ http://www.netasq.com
- IntruGuard http://www.intruguard.com
- Force10 http://www.force10networks.com
- RioRey http://www.riorey.com

3.2.1.1. Proventia

IBM Proventia Network IPS GX5208. Es un producto que permite protección preventiva para el núcleo de redes comerciales. Posee flexibilidad de puertos y una capacidad de análisis para tráficos de hasta 2 Gbps, cumpliendo satisfactoriamente con los requerimientos de las grandes empresas [51]. Proventia usa un sistema operativo basado en linux y asegurado de tal forma que solo se puede acceder a el a través de la interfase LMI o a través de la interfase serial. Las principales características del producto son:

- Arquitectura de administración escalable de tres niveles.
- Arquitectura de control de eventos y de red centralizada.
- Sensores en servidores y estaciones de trabajo.
- Análisis de seguridad y motor de base de datos centralizada
- Disposición de sensores simplificada.
- Interfase e usuario remoto segura y basada en roles.
- Vista lógicas “centrada en activos” de la seguridad de los datos.
- Control y comando de sensores por grupo.
- Análisis de datos orientado grupos.
- Soporte para otros productos ISS

- Integración con productos de terceras partes.
- "SecurityFusion" correlación en tiempo real y verificación de ataques.
- Módulo de administración de informes
- Procesos automáticos de actualizaciones de seguridad

La latencia del "Proventia" va de 82μs con una carga de 250Mbps y una longitud de paquetes de 512 bytes a un máximo de 198μs con with 1Gbps y una longitud de paquete de 128 bytes. [52]

3.2.1.2. Cisco IPS-4255 V5.0(3)

Este producto de Cisco provee detección pasiva IDS y IPS hasta velocidades de 500 Mbps. Cumpliendo los requerimientos de las empresas medianas. [53] El dispositivo cuenta con cuatro puertas de cobre que pueden ser configuradas con las velocidades 10710/1000 Mbps, y un puerto adicional de las mismas características para la administración. Es muy estable antes ataques prolongados de DOS y DDOS. Permite la administración a través de una interfase de linea de comandos o con una administración centralizada que permite administrar muchos dispositivos distribuidos a lo largo de la red.

Generador de Meta eventos (Meta Event Generator MEG) Cisco IPS incorpora a nivel de sensores correlación de eventos que provee a los administradores un método automatizado para mejorar el nivel de confianza de la clasificación de la actividad maliciosa detectada por los sensores. Este método provee un mecanismo que permite con las correspondientes acciones detener a gusanos y la inyección de vectores virósicos, como así también la propagación de gusanos. Esto es llevado a cabo mediante las siguientes técnicas:

- Correlación de alarmas de gusanos perteneciente a múltiples vulnerabilidades.
- Generación de meta eventos para secuencia de acciones conducentes a infestación de gusanos.

- Aumento automático de la severidad del ranking cuando grupos de eventos significa actividad de virus o gusanos.
- Mejoramiento de la fidelidad de las alarmas a través de disparos simultáneos basado en algoritmos de detección híbridos.

Nimda es un típico ejemplo de un gusano que utiliza múltiples vulnerabilidades durante su propagación a través de las redes. Las alarmas que se generarán por la utilización de esas vulnerabilidades serán disparadas en un muy corto intervalo de tiempo. Usando MEG el usuario puede especificar una secuencia lógica de eventos perteneciendo a ciertos gusanos en un solo meta evento llamado "Nimda". [54]

3.2.1.3. McAfee IntruShield 4010 V3.1.3

Esta basado en una mezcla de procesadores estándares y personalizados, el dispositivo McAfee IntruShield es un dispositivo de alto rendimiento ofreciendo detección y prevención de intrusiones contra ataques conocidos y desconocidos en tiempo real para redes corporativas. [55]

El sistema IntruShield es capaz de operar en redes a velocidades hasta de 2 Gbps, y es capaz de operar en linea o como IDS pasivo o ambos al mismo tiempo usando diferentes puertos en el mismo dispositivo. El producto incluye un sistema de administración llamado "IntruShield Security Management (ISM)" y los sensores cubren un rango de de ancho de banda desde 100Mbps a 2Gbps, permitiendo cubrir un escenario que va desde pequeñas empresas a grandes corporaciones.

Posee la capacidad de definir IPS virtuales esto permite definir políticas en forma separada para distintos niveles llegando hasta el nivel de servidor si se requiere.

El Firewall interno permite definir "Perímetros Virtuales" posee además el IntruShield capacidad de analizar tráfico encriptado SSL en tiempo real. El dispositivo cuenta con 12 puertos el cual permite un tráfico agregado de hasta 2 Gbps.

3.2.1.4. Secureworks isensor 850 V5.3

SecureWorks, Inc., fundada en 1999, es un proveedor de servicios de seguridad para internet a los fines de proteger redes corporativas contra ataques. SecureWorks brinda servicios para defender los activos informáticos de las empresas contra intentos de vulnerar la seguridad de esta. Los servicios de administración de seguridad ofrecidos por la empresa eliminan la necesidad de contar con personal especializado en seguridad. Los servicios disponibles incluyen detección de intrusiones basada en redes y basada en servidores. Permite un tráfico de red de hasta 100mbps. [56]

El detector de intrusiones en redes SecureWorks está integrado por los siguientes componentes: Network Intrusion Prevention Service is composed of the

- El iSensor, es un dispositivo de seguridad que monitorea el tráfico de la red y previene las intrusiones filtrando en forma activa los paquetes.
- El centro de Operaciones de Seguridad (Secure Operations Centre SOC), integrado por analistas de seguridad y operado las 24 horas del día los 7 días de la semana.
- Una serie de servidores que operan las redes y las aplicaciones del servidor que dan soportes a los iSensor, y la administración de la relación con los clientes y la infraestructura corporativa.

3.2.1.5. Juniper Networks IDP 600F V3.1

El Juniper Networks Intrusion Detection and Prevention IDP 600F es un dispositivo de detección de intrusiones que usa ocho métodos para detectar tráfico malicioso en la red. El IDP 600F es capaz de operar en ambos modos en linea como IPS y como detector de intrusiones pasivo conectado a un puerto espejo de un switch. El IDP 600F está diseñado para capturar y analizar tráfico a velocidades de redes de 500Mbps y con una muy baja latencia. [57] Usa una arquitectura basada en tres bloques: El sensor IDP, el servidor de administración y la interfase de usuario.

- El sensor IDP monitorea la red en la cual el IDP está instalado. Es un dispositivo de hardware propietario donde se ejecuta el programa IDP.

- El servidor de administración IDP Almacena y administra todos los objetos de ataque incluyendo firmas de ataques, anomalías de protocolos, información de registro, reglas básicas y políticas de protección. Múltiples sensores pueden ser administrados por un solo servidor de administración.

- Interfase de Usuario(UI). Consiste en una interfase gráfica desarrollada en Java para interactuar con el sistema IDP. La UI es usada para acceder en forma remota y manipular la información almacenada en el servidor de administración.

Existen tres implementaciones en linea:

- Modo puente transparente: En este modo el IDP es transparente y no necesita reconfiguración para que el servidor se percate de su presencia. Sin embargo el servidor no detecta al IDP en la red. Para el sistema IDP, el modo transparente es el más fácil de implementar y es el modo de operación recomendado. Usando soluciones de balance de carga de terceros, el IDP puede ser instalado en redes de alta disponibilidad y puentes en modo transparentes. El dispositivo bajo prueba es instalado en modo transparente.

- Modo Proxy ARP: En este modo la máquina servidora se percata de la presencia del IDP en la red y este automáticamente reconfigura la maquina servidora para enviar todos los paquetes que necesitan ser reenviados. El modo proxy ARP tiene mayor capacidad de envío de paquetes que el modo Puente transparente, pero solo funciona en redes en las cuales el tráfico es ruteado por un único router. El IDP puede ser instalado en forma independiente de la disponibilidad en configuración en modo proxy ARP.

- Modo Router: En este modo el IDP se comporta como un router típico, usando una tabla de ruteo para determinar donde los paquetes necesitan ser enviados, para encaminarlos al destino correcto. En una red típica el IDP debe ser configurado como la puerta de

enlace por omisión para todos los servidores protegidos. El modo router es tradicionalmente asociado con dispositivos de seguridad antiguos como firewalls, y solamente son ofrecidas estas opciones en IDP por compatibilidad. El IDP puede ser instalado en una configuración de alta disponibilidad usando soluciones de balance de carga de terceras partes o en una configuración independiente en el mismo router.

3.2.1.5.1. Máquina de detección El IDS opera como un dispositivo en linea que inspecciona todo el tráfico y determina que constituye una intrusión. Los mecanismos de detección pueden ser aplicados en forma bidireccional para detectar tanto ataques de tráfico cliente servidor como servidor cliente. La política de seguridad configurada en el IDP determina que acción tomara el IDP, descartar el paquete o solamente alertar.
Los métodos de detección usados son los siguientes:

- Firma con información de estado. El IDP incluye varios cientos de firmas de ataques conocidos. El administrador puede ver, editar o borrar firmas de ataques usando un editor en la UI. Firmas compuestas permiten englobar en un solo objeto compuesto múltiples firmas individuales y anomalías de protocolo, que permiten detectar ataques complejos en una sola sección El lenguaje para crear firmas personalizadas incluye el Perl y el estilo de expresiones regulares para facilitar la creación de firmas que pueden ser usadas para detectar una vulnerabilidad subyacente mas que un exploit basado en dicha vulnerabilidad. El número de firmas con información de estado cuando son escritas personalmente es mayor que 500.

- Anomalías de protocolo. Anomalías de protocolo son desviaciones de los paquetes de protocolo normalizado. La mayoría del tráfico legítimo está de acuerdo a lo establecidos por las RFC (Recuest For Comments) para los protocolos de Internet. El sistema IDP utiliza las RFC para crear objetos de anomalía de protocolo que se desvían de las especificaciones de protocolo publicadas. La cantidad de protocolos analizados por este IDP son más de 60.

- Backdoor (Puerta trasera) La detección backdoor usa patrones de tráfico de redes y heurística de la transmisión de paquetes para detectar tráfico interactivo, un signo común de un ataque usando Backdoor o Troyanos. Por el contrario los programas antivirus escanean el sistemas de archivos buscando archivos ejecutables en los servidores, este sistema IDP detecta el tráfico interactivo que es producido cuando los backdoors son usados, pero sin que se requiera una firma especifica. Este método asegura que el IDP puede detectar todos los backdoors que son usados aun aquellos que son desconocidos, y aún si los datos son encriptados. Tráficos normales bidireccionales tales como el FTP son indicados como normales según las políticas por defecto.

- Anomalías de tráfico. Una anomalía de tráfico es un patrón que indica actividad anormal del tráfico de la red, tales como búsquedas de puertos abiertos y otros ataques distribuidos. El IDP cuenta el número de puertos analizados en un específico lapso de tiempo y usa un análisis de flujo de tráfico para identificar búsquedas de puertos abiertos.

- IP Spoofing (Falseo de IP) El IP Spoofing ocurre cuando un paquete IP declara una dirección de origen falsa, el IDP verifica que la dirección de origen de un paquete entrante no pertenezca a una dirección de la red interna.

- Detección DOS. IDP detecta y previene ataques DOS tales como las inundaciones TCP SYN asegurándose que el saludo de tres vías TCP es realizado en forma correcta. En el modo en linea los requerimientos SYN son recibidos y recién son pasados a los servidores cuando estos se completan en forma satisfactoria. Cualquier requerimiento SYM incompleto es descartado después de un tiempo sin afectar a los servidores a los cuales el requerimiento está dirigido.

- Detección de ataques de capa 2. Los atacantes pueden manipular protocolos de capa 2 para realizar ataques ARP (Address resolution protocol) tales como envenenamientos del cache ARP y otros ataques MAC. El sistema IDP detecta ataques de capa 2 definiendo reglas implícitas en los sensores IDP. Reglas implícitas incluyen restricciones de las ta-

blas ARP administración de fragmentos, tiempo de espiración de las conexiones y otros mecanismos.

- Honeypot de redes. No es estrictamente un método de detección de intrusiones, el Honeypot de red IDP representa puertos abiertos en servidores existentes, para tentar a los atacantes a explotar las debilidades con el propósito de recoger evidencia.

3.2.1.6. Check Point IPS-1

Check Point IPS 1 es un sistema detector de intrusiones dedicado (IDS IPS) que ayuda a las organizaciones a asegurar sus redes corporativas, y proteger a los servidores críticos contra gusanos, malware automáticos y ataques combinados conocidos y desconocidos [58].

IPS 1 provee una seguridad fuerte robusta y dinámica. Además IPS-1 provee herramientas administrativas que incrementa la eficiencia de los administradores, proveyendo una rápida repuesta a los ataques. IPS-1 provee una interfase centralizada intuitiva y herramientas gráficas que permiten al administrador de redes identificar y reaccionar con rapidez ante los ataques a las redes. Además la administración de IPS-1 se encuentra integrada con el resto de los productos de Check Point, reduciendo costos de entrenamiento e incrementando la eficiencia de los operadores. Las soluciones IPS-1 están disponibles como equipo integral o software que se ejecuta en servidores abiertos. Check Point provee actualizaciones en tiempo real y boletines de alertas de seguridad provistos por SmartDefense Services, el cual es mantenido por Check Point Research Centers alrededor del mundo.

Beneficios

- Seguridad robusta: protege las redes contra ataques sofisticados y vectorizados.
- Administración eficiente.
- Implementación flexible

3.2.1.7. Radware

DefensePro [59] es un detector de intrusiones de múltiple capa que provee protección contra DoS, detección de anomalías y una gran variedad de ataques conocidos y desconocidos del día cero. Es fácil de usar y es una solución escalable para la protección contra virus y gusanos. La pro-actividad previene tanto ataques a servidores o redes, mientras asegura un alto rendimiento para el tráfico legítimo, aun bajo condiciones de ataque.
Como detector y prevención en linea, protector de DoS y conformador de tráfico, DefensePro esta diseñado para protección integral de las empresas y implementación en perímetros, centro de datos, campus universitarios, troncales de proveedores de servicios, integrando varios niveles de defensa, entre los que se incluye:

- Protección basado en firmas.
- Protección de irrupciones en servidores
- Mitigación de inundaciones DoS/DDoS
- Protección para ataques Encriptados SSL
- Control de acceso
- Administración de ancho de banda

DefensePro provee una seguridad sin precedentes integrando capacidades de protección basado en comportamiento adaptivo tanto a nivel de red y nivel de aplicación sin intervención de un operador.

DefensePro esta realizado con una arquitectura de hardware basado en ASIC (application specific integrated circuit Circuito integrado para aplicaciones específicas), que asegura el más alto nivel de seguridad disponibilidad y rendimiento. DefensePro puede monitorizar múltiples segmentos de red de núcleo y perímetro en ambientes de tráfico de hasta 6 Gbps.

DefensePro también se provee en una versión de software que puede ser ejecutado en una variedad de plataformas.

Máquina de decisión adaptiva

El mecanismo de auto aprendizaje basado en comportamiento, busca en forma pro-activa tráfico anómalo en la red, cuando DefensePro detecta un ataque caracteriza el comportamiento único del ataque, y establece un criterio de filtrado y ejecuta la contra medida apropiada. Un mecanismo de realimentación cerrado va modificando el criterio de filtrado en la medida que el ataque va mutando, protegiendo de esta forma la red contra ataques de alto grado de sofisticación en forma precisa. DefensePro es único en su habilidad para distinguir rápidamente entre tres amplias categorías de comportamiento: Tráfico legitimo y normal, trafico de ataque y trafico inusual creado por actividad legitima.

Principales características del DefensePro.

- Mantiene la continuidad del negocio aún cuando la red está bajo ataque.
- Asegura la continuidad y disponibilidad de las aplicaciones críticas (web, mail, FTP, DNS) bajo ataque
- Amplía cobertura contra ataques a la red incluyendo Gusanos, Troyanos, BOTs, pruebas de preataque y DoS.
- Bloque ataques sin bloquear el tráfico legitimo, servidores infestados pueden continuar trabajando en forma interrumpida.
- Reduce el costo total de administración de la seguridad.
- Se adapta a los cambios de las redes.
- Requiere una mínima configuración inicial, sin complicadas puestas a punto.
- Sencilla integración al ambiente de la red.
- Reduce el costo de enlace de los transportadores.

- Remueve el alto volumen de propagación de los gusanos e inundaciones DoS/DDoS.
- Rápida repuesta para ataques día cero sin bloquear el tráfico legitimo.
- Asegura el nivel de servicio contratado (SLAs) para los proveedores de servicio de Internet.
- Garantiza nivel de servicio usando reglas de administración de ancho de banda.

Sistema de administración y generación de reportes centralizado. Radware Insite provee una interfase de administración centralizada para toda su linea de productos.

3.2.1.8. SES

El DeepNines Security Edge System (SES) detiene ataques maliciosos de internet antes de que puedan llegar a la red corporativa. Esta diseñado como una defensa de primera linea para las redes, SES previene las brechas de seguridad que degradan la disponibilidad de las redes, restringiendo el uso de los recursos informáticos. [60]
SES es un dispositivo on line diseñado para proteger la red contra:

- Ataque basados en firmas posee mas de 8000 reglas.
- Ataques basados en anomalías del protocolo.
- Comportamiento anómalos 22 reglas heurísticas.
- Ataques de denegación de servicio y ataques de denegación de servicios distribuida.
- Ataques de auto propagación.
- Spyware
- Ataques de Phishing.
- Virus y Gusanos

SES protege el perímetro de la red detectando y bloqueando en forma inmediata múltiples ataques. Una vez que el tráfico deseado es permitido en la red, SES realiza una amplia inspección del tráfico de la red para asegurar que es tráfico deseado. Esta metodología propietaria reduce enormemente el número de firmas y políticas que deben administrarse mientras reduce la cantidad de falsos positivos y falsos negativos asociados con la mayoría de soluciones IPS. Construido con tecnologías bajo patentes SES US Patents 6,930,978 y 7,058,976 provee una solución poderosa. Entre las patentes de SES se pueden nombrar:

- Administración de tráfico inteligente. Permite un control total del tráfico y ante ataques que consuman recursos críticos disparar eventos para prevenir que los recursos del sistema se vean afectados.
- Tecnología de cero huella (Zero Footprint Technology ZFT la cual no permite que SES sea detectado desde la red evitando cualquier vulnerabilidad propia del mismo sistemas IPS.
- Inspección de paquetes profunda (Deep Packet Inspection DPI)

SES puede ejecutarse en cualquier equipamiento y soporta velocidades de hasta 1 Gbps inspeccionando todos los paquetes y secciones.

3.2.1.9. Enterasys

Enterasys Secure Networks es una empresa que abarca con sus productos todas las necesidades de una red corporativa, enrutadores, conmutadores, acceso inalámbrico y seguridad, posee clientes en setenta países y una variedad de productos que abarca desde la mediana hasta las grandes empresas, el producto de más altas prestaciones es el modelo Dragon cuyas principales características son las siguientes:.

- Velocidad de análisis 10 Gbps.
- Posee una de las base de datos de firmas más extensa del mercado aproximadamente 14000 firmas.

- Capacidad de análisis basado en Protocolo
- Capacidad de análisis basado en Anomalías
- Capacidad de análisis basado en Comportamiento

3.2.1.9.1. Arquitectura del IDS/IDP Posee una arquitectura distribuida compuesta de los siguientes elementos.

- Sensores de red
- Sensores de servidores
- Servidor de Administración
- Procesadores de flujo de eventos

3.2.1.10. Securesoft

Es una empresa japonesa dedicada exclusivamente a la seguridad posee una amplia gama de productos IPS, que permite ajustar el equipamiento a las necesidades específicas de la empresa. Los principales productos IPS son:

- Sniper IPS 4000
- Sniper IPS 2000
- Sniper IPS 1000
- Absolute IPS NP5G 5 Gbps
- Absolute IPS 1000 1 Gbps
- T series

Securesoft centra su mercado en Japon y China ya que su página web posee versiones solo en Japones y Chino.

3.2.1.11. Toplayer

TopLayer es una empresa dedicada exclusivamente a la fabricación de equipos IPS, con desarrollos de hardware propios, con arquitecturas basados en ASIC y FPGA logrando una muy baja latencia.

Uno de los modelos más destacados es la serie Top Layer IPS 5500 la cual tiene una capacidad de análisis de tráfico desde 600 Mbps hasta 4.4 Gbps. Baja Latencia >50 uS. lo que permite manejar tráfico de alta sensibilidad a la latencia como VOIP.

Basado en ASIC y escalable permitiendo configurar cluster de hasta ocho Top Layer 5500 para alcanzar mayores velocidades de tráfico de hasta 10 Gbps.

Cumple con los requerimientos PCI-DSS.

La serie E posee una arquitectura de validación de archivos bastante interesante que combina Módulos de validación de protocolos con módulos de validación de datos de la carga útil.

Se puede distribuir los módulos Top Layer 5500 a lo largo de toda la red y administrarlos en forma centralizada.

3.2.1.12. Nitrosecurity

Nitrosecurity es una empresa dedicada a la seguridad de sistemas su principal producto IPS es el Nitro Guard 4000 IPS con una capacidad de análisis de 1,5 Gbps, esta basado en Snort desarrollando su propia máquina de detección que supera la original de Snort con un conjunto de firmas de ataques propio de mas de 4500 firmas.

Las características mas importantes de Nitro Guard 4000 son las siguiente:

- Análisis de flujo de red sFlow y netFlow
- Excelente consola de administración NitroView
- Administrador de eventos de seguridad de la información (Security Information Event Management SIEM).
- Esta incluida en las 5000 empresas con mayor crecimiento en el 2007

- Elegida por la U.S. Army en el 2008.
- Bajo Costo por Gbps

Es de notar que esta firma es la que desarrolló Snort y sigue manteniendo el producto bajo liencia GNU.

3.2.1.13. Broadweb

Broadweb es una firma dedicada a la seguridad de redes con una amplia gama de productos IPS. Esta firma fabrica procesadores para detección de intrusiones para sus propios productos y para terceras partes.
Entre la serie de productos IPS se encuentran:

- OneKeeper series. Son dispositivos que integran firewall, VPN, IPS, traffic shaping, antivirus, filtro de contenidos y antispam.
  Permite un análisis de tráfico a una velocidad de 100 Mbps.
- NK6000 Es un equipo específicamente diseñado para IPS posee una base de datos de 17000 firmas y una capacidad de análisis de tráfico a una velocidad de 1 Gbps, esta orientado a carriers y grandes empresas.
- NK5100 series Es un equipo IPS orientado a empresas medianas e incluye además toda la protección que puede necesitar una empresa mediana como:
  - Anti - Intrusion
  - Anti - DoS / DDoS
  - Anti - P2P
  - Anti - Instant Messenger
  - Anti - Web Post
  - Worm - Mitigation

- NK3000 Es un equipo IPS orientado a empresas pequeñas y medianas e incluye además toda la protección que puede necesitar las pequeñas empresa o sucursales regionales como:
    - Ataques de Buffer Overflow
    - Ataques Barridos de puertos
    - Ataques Troyanos
    - Ataques de fragmentación IP
    - Virus y Gusanos
    - Ataques a vulnerabilidades de sistemas y aplicaciones
    - Ataques de Denegación de Servicio DoS/DDoS
    - P2P, IM, Spam, Web port, Sitios pornográficos
- NK Eulen Es es un equipo IPS orientado a pequeñas empresas y sucursales regionales e incluye además toda la protección que puede necesitar una empresa mediana como:
    - Anti - Intrusion
    - Anti - DoS / DDoS
    - Anti - P2P
    - Anti - Instant Messenger
    - Anti - Web Post
    - Anti - Web mail
    - Firewall basado en políticas

Entre los procesadores se encuentran los siguientes:

- BS2oC Es un procesador ASIC orientado a la construcción de sistemas IPS y UTM.

3.2.1.14. Fortinet

Las soluciones IPS están alrededor de los ATCA (Advanced Telecom Computing Architecture) Arquitectura avanzada de computo de telecomunicaciones. La serie de productos esta orientada a empresas de telecomunicaciones proveyendo IPS y UTM, los principales productos son:

- Fortigate 5140 con una capacidad de tráfico de 70 Gbps
- Fortigate 5050 con una capacidad de tráfico de 25 Gbps
- Fortigate 5020 con una capacidad de tráfico de 10 Gbps
- Fortigate 5005 con una capacidad de tráfico de 5 Gbps
- Fortigate 5002 con una capacidad de tráfico de 2 Gbps
- Fortigate 5001 con una capacidad de tráfico de 1 Gbps

Fortinet mantiene una base de datos actualizada de mas de 60000 amenazas. Los equipos están desarrollados con una arquitecta propia basados en ASICs propietarias.

3.2.1.15. Sourcefire

La empresa Source es la creadora del programa open source Snort y ClamaAV, posee una serie de productos IPS Sourcefire 3D de esta serie el producto más destacados es el Sourcefire 3D9800 permitiendo un tráfico de 10 Gbps. El sistemas 3D pude ejecutar los siguientes módulos:

- Sourcefire IPS (Intrusion Prevention System) provee lo mejor en su clase reforzando la potencia del estándar de la industria que es el Snort y la máquina de detección basadas en las reglas de Snort. Soportado por el conocido equipo de investigación de vulnerabilidades (Vulnerability Research Team VRT).

- Sourcefire Alerta de redes en tiempo real (RNA Real-time Network Awareness) monitorea la red en tiempo real en forma pasiva proveyendo inteligencia de la red incluyendo los sistemas operativos, servicios, aplicaciones y protocolos y las posibles vulnerabilidades que puedan existir en la red. RNA automatiza las funciones claves mientras provee potencialidades superiores al manejo de incidentes incluyendo visibilidad de la red y análisis de comportamiento.

- Sourcefire Alerta de usuarios en tiempo real (RUA Real-time User Awareness) verifica Directorios activos y nombres de usuarios LDAP con direcciones IP de servidores involucrados en la seguridad.

- Sourcefire Análisis de flujo (NetFlow Analysis) es un componente adicional de las soluciones de Sourcefire. Con el agregado de Análisis de Flujo de router y conmutadores aborda la cobertura de la seguridad interna para incidentes no cubiertos y medios para evaluar la provisión de ancho de banda y diagnósticos de la red.

Estos módulos proveen defensa contra las siguientes amenazas.

- Gusanos
- Trojanos
- Búsqueda de Puertos
- Ataques de Buffer overflow
- Ataques de denegación de servicio
- Anomalías de Protocolo
- Trafico Mal formado
- Encabezamientos inválidos
- Ataques a VoIP

- Ataques a IPv6
- Ataque de fragmentación y evasión
- Ataques del día Cero

3.2.1.16. Stonesoft

Productos

- StoneGate IPS-6100 4Gbps
- StoneGate IPS-6000 2Gbps
- StoneGate IPS-2000 600Mbps
- StoneGate IPS-400 100 Mbps
- StoneGate SGI-2000S 1200 Mbps
- StoneGate SGI-200S 400 Mbps
- StoneGate SGI-200C 400 Mbps
- StoneGate SGI-200N 400 Mbps
- StoneGate SGI-200ANZ 400 Mbps
- StoneGate SGI-20A 80 Mbps

3.2.1.17. Tipping Point

3.2.1.18. Reflex Security

Modelos hasta 10 Gbps Dispositivos de seguridad virtualizados Se apoya en la base de datos de firmas Snort.

3.2.1.19. Still secure

Es una empresa dedicada a productos IPS los modelos cubren las necesidades de pequeñas y medianas empresas además posee un programa IPS (Strata Guard) Basado en Snort con distribución gratuita. Los principales productos de Still secure son:

- SMB 10 Mbps
- Enterprice 200 Mbps
- GigE 1 Gbps
- Ofrece el Strata Guard para 5 Mbps gratis.

3.2.1.20. TrustWave

TS 1000 Velocidad 1.3 Gbps Latencia 30 uS

3.2.1.21. NETASQ

Productos

- UTM U1100 2,8 Gbps
- UTM U1500 3.8 Gbps
- UTM U 6000 5 Gbps

3.2.1.22. IntruGuard

Productos

- IG200 0.1 Gbps
- IG2000 2 Gbps

Usa ASICS a medida Latencia <50 uS

3.2.1.23. Force10

Serie P Velocidad 10 Gbps 1 Mpps Latencia <2 uS

3.2.1.24. RioRey

Productos

- RX1200
- RX2300
- RX3300

3.2.2. Productos IDS GNU

3.2.2.1. Shadow

El proyecto Shadow (www.nswc.navy.mil/ISSEC/CID) es un desarrollo conjunto de Centro de Armas Navales “Dahlgren", de la Agencia Nacional de seguridad (NSA) y del instituto SANS. Shadow usa estaciones de censado y de análisis. Los sensores usualmente se hallan ubicados en puntos claves de la red, tales como puntos fuera del firewalll, mientras que las estaciones de análisis se encuentran en la parte interna del firewalll. Los sensores están basados en programas de dominio publico de captura de paquetes y estos no procesan los datos, de esta forma evita que un intruso determine los objetivos de detección capturando un sensor desprotegido. Los sensores extraen el encabezamiento de los paquetes y los almacenan en un archivo que la estación de análisis, lee en forma periódica. El principal soporte es de tcpdump, filtros de paquetes suple-mentado por herramientas basadas en perl para detectar intrusiones lentas que pueden propagarse a múltiples archivos de registro. La estación de análisis usa una interfase basada en web para mostrar los resultados, como así también los datos en crudo. Shadow se puede ejecutar en cualquier sistema Unix incluido FreeBSD y Linux.

3.2.2.2. Blare

Blare es un detector de intrusiones basado en políticas corre sobre plataforma Linux, el principal propósito es servir como banco de pruebas para experimentar con nuevos sistemas de detección de intrusiones.

A diferencia de otros sistemas de detecciones como Snort, Blare no requieres firmas de ataques, o perfiles de aprendizaje ni conocimiento de los programas atacantes sus principales objetivos son:

- Detectar todas las violaciones a las políticas de seguridad, incluyendo violaciones usando ataques nuevos y desconocidos.
- Reportar solo las violaciones a las actuales políticas de seguridad, sin falsos positivos.
- Permite trabajar con políticas usuales de seguridad tales como acceso discrecional, Bell-LaPadula etc.

3.2.2.3. Bro

Bro [61][12][45] es un detector de intrusiones en redes Open Source basado en el sistema operativo UNIX, monitorea en forma pasiva el tráfico de la red e inspecciona por actividad sospechosa. Bro detecta intrusiones en primer lugar analizando el tráfico para extraer información semántica a nivel de aplicación, y entonces ejecutando analizadores orientados a eventos, que comparan la actividad con firmas conocidas. Este análisis incluye la detección de ataques específicos definidos por sus firmas y actividades inusuales, tales como intentos de conexiones fallidas a ciertos servicios. Bro utiliza un lenguaje especializado de políticas que permite configurar la operación de Bro, tanto si evolucionan las políticas del sitio o son descubiertos nuevos ataques. Si Bro detecta algo de interés, este puede ser instruido, para generar un entrada en el registro, alertar al operador en tiempo real, o ejecutar un comando del sistema, de esta forma ante un ataque Bro puede bloquear instantáneamente el ataque y la información de registro puede ser de ayuda para la el análisis forense. Bro esta orientado a la detección de intrusiones

de sistemas de alto volumen de tráfico (1 Gbps). Realizando tareas a nivel de filtrado de paquetes, logrando un rendimiento satisfactorio, ejecutándose en Computadoras comerciales (PC) de bajo costo logrando una gran relación rendimiento costo. Bro ha sido desarrollado como una plataforma para experimentar con nuevos métodos de detección de intrusiones y análisis de trafico, y orientado a expertos en UNIX y no a quienes quieren una herramienta lista para usar.

3.2.2.4. Snort

Snort [11] es un IDS o Sistema de detección de intrusiones basado en red (NIDS). Implementa un motor de detección de ataques y barrido de puertos que permite registrar, alertar y responder ante cualquier anomalía previamente definida como patrones que corresponden a ataques conocidos, barridos, intentos de aprovechar alguna vulnerabilidad, análisis de protocolos, etc. Este programa realiza todas estas tareas en tiempo real.

Snort (http://www.snort.org/) está disponible bajo licencia GPL, gratuito y funciona bajo plataformas Windows y UNIX/Linux. Es uno de los más usados y dispone de una gran cantidad de filtros o patrones ya predefinidos, así como actualizaciones constantes ante casos de ataques, barridos o vulnerabilidades que vayan siendo detectadas a través de los distintos boletines de seguridad.

Este IDS implementa un lenguaje de creación de reglas flexibles, potente y sencillo. Durante su instalación nos provee de una gran variedad de filtros o reglas para backdoor, ddos, finger, ftp, ataques web, CGI, búsquedas Nmap.

Puede funcionar como sniffer (podemos ver en consola y en tiempo real qué ocurre en nuestra red, todo nuestro tráfico), registro de paquetes (permite guardar en un archivo los registros para su posterior análisis, análisis (“offline”) o como un IDS normal (en este caso NIDS).

La colocación de Snort en nuestra red puede realizarse según el tráfico que se desea detectar: paquetes entrantes, paquetes salientes, dentro del firewall, fuera del firewall proveyendo una arquitectura de defensa muy flexible.

Una característica muy importante e implementada desde hace pocas versiones es FlexResp. Permite, dada una conexión con tráfico malicioso, darla de baja, esto es hacerle un DROP

mediante el agregado de una regla en el Firewall de tal forma que envié un paquete con la bandera RST activa, con lo cual cumpliría funciones de IPS (Intrusion Prevention System), cortando las conexiones que cumplan ciertas reglas predefinidas. No sólo corta la conexiones ya que puede realizar otras acciones adicionales.

3.2.3. Evaluación de IDS

La evaluación de productos IDS e IPS ha sido tomada de la publicación de Gartner [62] en este análisis la empresa fija ciertos criterios para la inclusión de los productos en este análisis comparativo:

- Operación en linea a la velocidad de transferencia de la linea física.
- Realiza normalización armado e inspección de paquetes
- Aplica reglas basada en varios metodologías de análisis del flujo de paquetes tales como anomalía de protocolos, análisis de firmas, y análisis de comportamiento.
- Descarte de secciones maliciosas y el descarte no debe bloquear el tráfico siguiente. Haber vendido en el mercado productos IPS durante el ultimo año por más de U$S 4 millones de dolares.

Criterios de evaluación Los criterios de evaluación fijados por Gartner [63] [62] se pueden resumir en la Figura 3.1 a continuación se explicita los distintos criterios.

- El criterio en la capacidad para ejecutar se basa en:
    - Producto y Servicio: Esto incluye la satisfacción del cliente en cuanto al despliegue; Se asigna altas calificaciones de acuerdo al rendimiento demostrado en evaluaciones de competitividad, entre los mejores de su clase y la calidad de detección de firmas.
    - Viabilidad general: Esta se ocupa de la liquides financiera de las empresas y de las perspectivas de la continuidad de las operaciones.

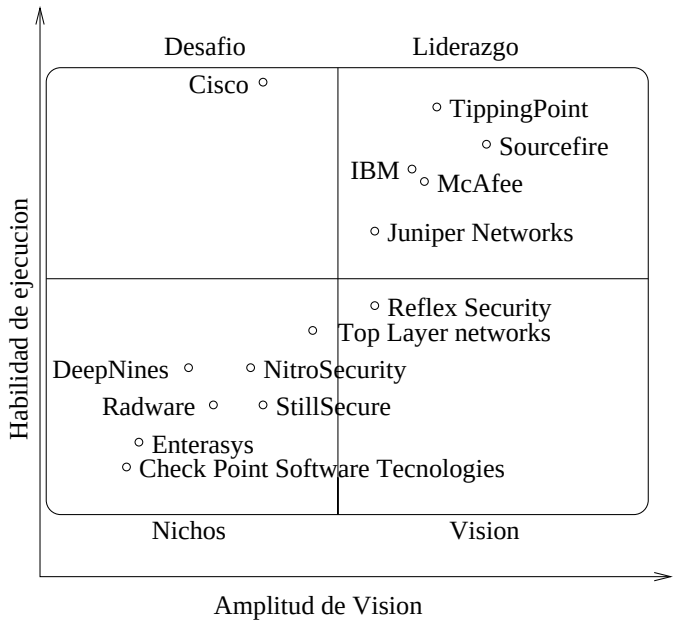

Figura 3.1: Cuadrante de Comparación de IPS.

- Ejecución de venta / Precios: Esto incluye dólares por Gbps, ingresos, el volumen medio por operación, base instalada y uso por proveedores de administración servicios de seguridad (Managed Security Service Providers MSSPs).
- Repuesta al Mercado: Esto significa la respuesta de la empresa a las nuevas funciones previstas.
- Ejecución del mercado: Esto incluye la entrega de las características requeridas y el rendimiento, la satisfacción del cliente con las características y el posicionamiento ante la elección del cliente frente a los competidores. Se le asignó alto puntaje en esta categoría a los vendedores que ofrecen productos con baja latencia y multi-Gbps, poseen una sólida seguridad interna, como se comportan ante ataques, tienen una alta disponibilidad y están disponibles los puertos que cumplan con la demanda de los clientes. También son altamente valorados los vendedores cuyos productos ofrecen una alta velocidad de producción de firmas de vulnerabilidades, la calidad de la firma y aquellas empresas que dedican los recursos internos al descubrimiento de nuevas vulnerabilidades.
- Experiencia del Cliente: Esto incluye la gestión de la experiencia y trayectoria, y

Cuadro 3.1: Capacidad para ejecutar los criterios de evaluación.

Criterio de evaluación	**Peso**
Producto y Servicio	Alto
Viabilidad en general	Normal
Ventas y precios	Normal
Repuesta al mercado	Normal
Ejecución del mercado	Normal
Satisfacción del cliente	Normal
Operaciones	Normal

la profundidad de la experiencia personal, específicamente en el mercado de seguridad. También es importante baja latencia, rápida actualizaciones de firmas, en general baja tasa de falsos positivos y falsos negativos, y el comportamiento del producto en el ataque. y la satisfacción del cliente posterior al ataque, estos son criterios claves.

- Operaciones: Esta es la capacidad de la organización para cumplir sus metas y compromisos. Los factores que incluyen la calidad de la estructura orgánica, incluyendo habilidades, experiencias, programas, sistemas y otros vehículos que permitan a la organización operar de manera eficaz y eficiente en forma permanente (véase el cuadro 3.1).

- Los criterios de cobertura de la visión incluyen:

  - Comprensión del mercado: Esto incluye proporcionar la correcta combinación de detección y bloqueo de las tecnologías que deben cumplir y exceder las necesidades de los clientes. También se incluyen una comprensión y un compromiso con el mercado de la seguridad y más concretamente, el mercado de la seguridad en las redes.

  - Estrategia para el mercado: Este criterio incluye la innovación, la previsión de las necesidades de los clientes, habiendo una vulnerabilidad en lugar de centrarse en

la explotación del producto, estar por delante de los competidores para desarrollar nuevas características y la integración con otras soluciones de seguridad. Los vendedores que dependen de terceros para fuentes de firmas o que tienen escasos recursos para detectar las tecnologías de detección tienen la puntuación más baja.

- Estrategia de venta: Esto incluye pre y post soporte de producto, valor para la fijación de precios y el suministro de explicaciones claras y recomendaciones para la detección de eventos.
- Oferta (Producto) Estrategia: Este criterio hace hincapié en la hoja de ruta de los productos, calidad de firma, NGFW (Next Generation Firewall) integración y rendimiento. Completar con éxito pruebas de terceros, tales como el NSS Lab [64], grupo de pruebas IPS y criterios comunes de las evaluaciones, son importantes. Los vendedores que reusen firmas, dependan demasiado en la detección de comportamiento y no posean firmas de buena calidad no corresponde una buena puntuación.
- Modelo de Negocios: Esto incluye el proceso y la tasa de éxito para el desarrollo de nuevas características y la innovación, gasto en I + D.
- Vertical / Industria Estrategia: Este criterio incluye la capacidad de dirigir los recursos, capacidades y ofertas para satisfacer las necesidades específicas del mercado y un compromiso con los mercados verticales (por ejemplo, MSSP y el sector financiero).
- Innovación: Esto incluye la I + D y diferenciadores de la calidad, como el rendimiento, la interfaz de administración y la claridad de la presentación de informes. La hoja de ruta debe incluir el desplazamiento de los dispositivos IPS a nuevos puntos de posicionamiento y mejor desempeño de los dispositivos, así como técnicas avanzadas para detectar y bloquear los ataques dirigidos.
- Estrategia geográfica: Esto incluye la capacidad y el compromiso de orientar recursos para satisfacer las necesidades específicas de las zonas geográficas fuera del mercado local en forma directa o a través de socios comerciales, canales de distri-

Cuadro 3.2: Cobertura de la Visión de los criterios de evaluación.

Criterio de evaluación	**Peso**
Comprensión del mercado	Normal
Estrategias de mercado	Bajo
Estrategias de venta	Bajo
Oferta	Alto
Modelo de negocios	Normal
Estrategia Industrial	Bajo
Innovación	Alto
Estrategia geográfica	Alto

bución y filiales, según proceda, para la geografía y el mercado (véase el cuadro 3.2).

- Liderazgo.

  Deberá demostrar un progreso y esfuerzo equilibrado en todas las ejecuciones y la visión de las categorías. Sus acciones deberán estar orientadas a aumentar la competitividad de todos los sectores y para todos los productos del mercado, y que pueden cambiar el curso de la industria. Para seguir siendo líderes, los proveedores deben haber demostrado un historial de éxito en la entrega de IPS a las empresas, el despliegue y las evaluaciones de competitividad en casos de éxito. Líderes de producir productos que proporcionan alta calidad de firmas, baja latencia, están innovando con o delante de los clientes retos (como el uso variable de inteligencia para hacer más eficientes las detecciones) y tienen una amplia gama de modelos. Líderes ganar continuamente las selecciones de los clientes y son visibles constantemente en las listas de empresas lideres. Sin embargo, un proveedor líder no es una elección por defecto para cada comprador, y los clientes no deben asumir que tienen que comprar sólo de los proveedores líderes en el cuadrante.

- Desafíos

  Desafíos que tienen los productos típicos de abordar las necesidades del mercado, con fuertes ventas, la visibilidad y la influencia que se suman a una mayor ejecución de

jugadores de nicho. Desafíos tienen éxito en establecer las bases de los clientes pero fallan en la selección competitiva.

- Visionarios

  Visionarios invierten en las principales características de punta que serán significativas en la próxima generación de productos y que proporcione a los compradores acceso temprano a la mejora de la seguridad y la gestión. Visionarios pueden afectar el curso de la evolución tecnológica en el mercado, pero carecen de habilidades para la ejecución y desafíos de los líderes del mercado.

- Nichos de mercado

  Los Nichos de mercado ofrecen soluciones viables que responden a las necesidades de algunos compradores. Los Nichos de mercado tienen menos probabilidades de aparecer en las listas de lideres, pero la tarifa y cuando se le da el derecho de oportunidades. Aunque por lo general carecen de influencia para cambiar el curso del mercado, que no debe considerarse simplemente como los siguientes referentes del mercado. Los Nichos de mercado podrán dirigirse a subconjuntos del conjunto del mercado (por ejemplo, las pequeñas o medianas empresas [PYME] o un segmento de mercado vertical), y a menudo lo hacen de manera más eficiente que los líderes. Los Nichos de mercado con frecuencia son las empresas más pequeñas, sólo producen el software de los aparatos y aún no tienen los recursos para satisfacer todas las necesidades de la empresa.

Capítulo 4

Estadística

4.1. Introducción

Los métodos estadísticos han sido utilizados en sistemas de detección de intrusos para detectar actividades anómalas en una red, sin embargo el nivel de análisis se basa, en el cálculo de medias y varianzas de algunas variables y detectan si se excedieron de ciertos niveles considerados como tráfico normal. Otros métodos se basan en correlacionar tráfico normal con tráfico bajo ataque un ejemplo es el trabajo de Jao B. D. Cabrera, B. Ravichandran y Raman K. Mehra [65]. Desarrollos más avanzados utilizan test de $\chi 2$ (CHI cuadrado) para medir similitud entre diferentes intervalos de tiempo. La estadística de Kolmogorov-Smirnov es usada en este trabajo de investigación y en sistemas similares a este para modelar y detectar patrones de tráfico. A continuación con el objetivo de mostrar las distintos herramientas estadísticas que utilizan los diferentes sistemas IDS se presenta un resumen de estos métodos.

4.2. Detección de ataque por correlación de variables

Este método se basa en realizar una correlación de variables claves durante funcionamiento normal de la red y bajo ataque.

En primer lugar se deben elegir variables claves, un método es basarse en el conocimiento

del dominio del tipo de ataque a detectar; Por ejemplo en un ataque de inundación de ping la variable MIBS “icmpInEchos” seria una de las apropiadas a considerar. Una segundo método puede ser comparar la evolución de cada variable durante la operación normal y durante el ataque y la variable que experimente mayores variaciones seria la variable clave a elegir, a su vez que se busca variaciones de variables las series de tiempo se deben subdividir en lapsos menores llamadas ventanas de observación..

4.3. Generación predictiva de patrones

4.3.1. Propósito

La detección de intrusiones basadas en reglas (Rules Based Intrusion Detection Systems RBIDS) tiene el inconveniente de que se basa en reglas conocidas, no pudiendo detectar detectar intrusiones nuevas o variantes de intrusiones conocidas para superar este inconveniente se recurre a métodos estadísticos para elaborar perfiles de ataques en forma predictiva (SBIDS) [66] que sirvan de patrones para comparar con el tráfico de la red y así detectar comportamientos anormales que constituyan una amenaza a la seguridad de las redes.

4.3.2. Detalles técnicos

Los sistemas SBID buscan identificar comportamientos abusivos analizando datos que se desvíen del comportamiento normal esperado, estos sistemas se basan en la premisa que las intrusiones se comportaran significativamente diferente del tráfico normal. Antes de que una actividad inusual pueda ser detectada es necesario la caracterización del tráfico normal, llamado perfil de tráfico que es una secuencia de eventos que se encuentra en los sistemas de auditoría de la red. Cualquier secuencia de eventos que se aleje de este comportamiento de una forma estadísticamente significativa es marcada como intrusión. Una de las principales ventajas de los sistemas SBIDS consiste en que las intrusiones pueden ser detectadas sin una base de datos de firmas elaboradas a priori.

Variaciones interesantes de este método incluyen las siguientes:

- Generación predictiva de patrones, la cual usa regla basadas en perfiles de comportamiento definidas como secuencias de eventos con pesos estadísticos, la ventaja de este método es que detecta tanto intrusiones como mal uso de la red.
- Aproximaciones basada en conexiones. hace uso de redes neuronales para mantener los perfiles de comportamiento.

4.3.2.1. Madurez

Los algoritmos para detectar intrusiones en redes se usan desde 1988. a partir de esa fecha se han desarrollado muchos prototipos de sistemas entre los cuales se pueden mencionar Haystack [Smaha 88], IDES [Lunt 93] y el proyecto MIDAS
MIDAS es un SBIDS en tiempo real desarrollado para proveer seguridad al centro Nacional de Computación en Estados Unidos este esta implementado en SRI y en el FBI combinando SBIDS con RBIDS para detectar un amplio rango de intento de intrusiones.
Otra implementación de SBIDS es la de AT&T Bell Lab's el sistema Dragons el cual protege la puerta de enlace de Internet detectando intentos de intrusiones.

4.3.3. Costos y Limitaciones

Adicionalmente a los costos asociados a mantener los archivos de auditoría y los perfiles de usuarios hay riesgos y limitaciones en el uso de la tecnología SBIDS:

- Dado que los perfiles de usuarios son puestos al día periódicamente, es posible para un usuario interno ir modificando paulatinamente su perfil hasta que un ataque pueda ser realizado sin que el sistema se de cuenta.
- Determinando un apropiado nivel para una desviación estadística significativa puede ser difícil de lograr. Si el nivel se elige demasiado bajo, actividades anómalas no intrusivas son detectadas provocando falsos positivos, por otro lado si se elige el nivel muy alto actividades anómalas intrusivas no son detectadas provocándose falsos negativos.

Cuadro 4.1: Matriz de entrada.

	Ignorar	**Comprobar**
FP	A_1	$-A_2$
TP	$-B_1$	B_2

- Definir perfiles de usuarios puede ser difícil, especialmente para usuarios con horarios, y hábitos de trabajo errático.

4.4. Información FPP basada en redes bayesianas

.Cuando se procesa estadísticamente muestras de paquetes de la red se debe conocer cuantas muestras se deben almacenar para garantizar la seguridad de la red. para resolver este problema se debe desarrollar el concepto de probabilidad de falso positivo (FPP False Positive Probability), entonces se construye un modelo para los alertas IDs basados en decisión estadística, y demostrando la existencia de un nivel de FPP, que puede garantizar la operación segura.

4.4.1. Modelo de trabajo

Suponiendo que las entradas provenientes de alertas tienen igual peso, se puede definir una matriz de de decisión estadística de acuerdo a la tabla 4.1 donde A_1, A_2, B_1, B_2 son mayores que 0, y representan las medias de la acción de cada entrada.

Se define la probabilidad de falso positivo (FPP) como:

$$p = P\left(alert = FP\right) \tag{4.1}$$

La probabilidad de ignorar una alerta bajo un índice de seguridad α es marcado como $P\left(\alpha\right)$. Entonces se puede calcular la entrada esperada para cada acción :

$$E\left[\Omega/ignorar\right] = pA_1 - (1-p)B_1 = p(A_1 + B_1) - B_1 \tag{4.2}$$

$$E\left[\Omega/comprobar\right] = pA_2 + (1-p)B_2 = B_2 - p(A_1 + B_2) \tag{4.3}$$

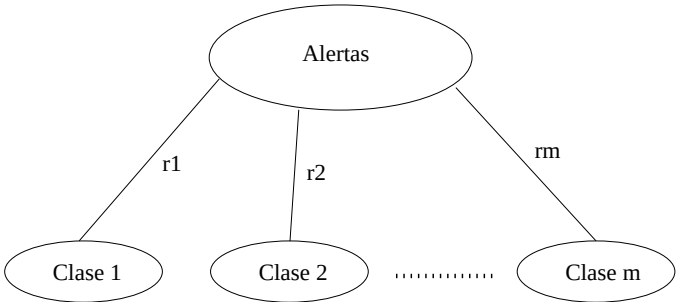

Figura 4.1: Clasificación de alertas.

$$E[\Omega] = P(\alpha)E[\Omega/ignorar] + (1 - P(\alpha))E[\Omega/comprobar] \tag{4.4}$$

Suponiendo que el número de alertas es N, el número de muestras debe ser n, entonces la variable estocástica n obedece a la distribución de Bernoulli $B(N, p)$. Si el sistema de seguridad tiene una confianza α. Entonces el número de muestras n debe satisfacer la desigualdad $p^n \leq \alpha$ la cual se puede expresar como:

$$n \leq \frac{\log \alpha}{\log p} \tag{4.5}$$

a valores más chicos de α significa mayores requerimientos de seguridad de la ecuación 4.5 se encuentra que los números de alertas a comprobar serán mayores si la probabilidad de falsos positivos se vuelve mayor y los requerimientos de seguridad crecen.
Y la pérdida por ignorar una alerta es :

$$E[\Omega/ignorar] = N(1-p)B_1 \tag{4.6}$$

Y la pérdida por comprobar una alerta es :

$$E[\Omega/comprobar] = nB_2 = B_2 \frac{\log \alpha}{\log p} \tag{4.7}$$

Se pueden clasificar los alertas en m clases de acuerdo a alguna norma; $clase_1, clase_2 ..... clase_m$ los porcentajes de cada clase son $r_1, r_2 ..... r_m$ y la probabilidad de falsos positivos esta representada por $p_1, p_2 ..... p_m$ cuando se construye una red de probabilidad de falsos positivos tiene una estructura como se muestra en la fig 4.1

Cuadro 4.2: Tabla de contingencias observadas para (A,B).

	B	$\overline{B}$
A	$nP(A \cap B)$	$nP(A \cap \overline{B})$
$\overline{A}$	$nP(\overline{A} \cap B)$	$nP(\overline{A} \cap \overline{B})$

De acuerdo a la formula de probabilidad total, la probabilidad total de falso positivo es:

$$p = \sum_{i=1}^{m} r_i p_i \tag{4.8}$$

Se pueden clasificar las alertas de cada clase por otra norma. Suponiendo la $clase_i (i = 1, ...m)$ es clasificada en la $clase_{i1}, clase_{i2}.....clase_{im_i}$ el porcentaje de alertas en cada clase para alertar a su clase padre esta dada en orden: $r_{i1}, r_{i2}.....r_{im_i}$ Entonces la probabilidad total de falso positivos es:

$$p = \sum_{i=1}^{m} r_i \sum_{j=1}^{m_i} r_{ij} p_{ij} \tag{4.9}$$

Si la probabilidad de falso positivo de cada clase es conocido mediante esta forma se puede calcular la probabilidad de falso positivo total.

4.5. Detección de intrusiones mediante la distancia chi-cuadrado.

Xin Jin desarrolla la técnica estadística de chi-cuadrado para detección de intrusiones en redes [67]. La técnica estadística de chi-cuadrado se utiliza para medir el grado de dependencia entre dos variables (A,B) [68]. Para calcular la estadística de chi-cuadrado de dos variables (A,B) se requiere construir dos tablas de contingencia. La tabla para la contingencia observada (A,B) tiene cuatro celdas correspondiendo a los cuatro valores booleanos posibles de las combinaciones A, B. El valor en cada celda es el número de muestras que coinciden con la combinación booleana para cada celda. Esos valores pueden ser expresados en términos del número total de muestras n y de las frecuencias relativas observadas correspondientes a las cuatro combinaciones booleanas como se muestra en la tabla 4.2 El análisis de chi-cuadrado

Cuadro 4.3: Tabla de contingencias esperada para (A,B).

	B	$\overline{B}$
A	$nP(A)P(B)$	$nP(A)(1-P(B))$
$\overline{A}$	$n(1-P(A))P(B)$	$n(1-P(A))(1-P(B))$

requiere que la tabla de contingencia observada debe ser comparada con la tabla que se obtiene si se hace tender el número de muestras asintoticamente a infinito ($n \to \infty$) como si las variables A,B fueran estadísticamente independientes, como se muestra en la tabla 4.3.

La estadística de chi-cuadrado es definida en términos de las entradas de la tabla de contingencias tabla 4.2 y la tabla de las contingencias esperadas tabla 4.3 mediante la siguiente formula:

$$\chi^2 = \sum_{0 \leq i,j \leq 1} \frac{\left(observado_{i,j} - esperado_{i,j}\right)^2}{esperado_{i,j}} \tag{4.10}$$

Por lo tanto χ^2 representa la suma de la desviación cuadrada de los valores observados de los valores esperados. El nivel de significancia estadístico correspondiendo a valores específicos de χ^2 pueden ser encontrados en las tablas de distribución de χ^2. Para la detección de intrusiones las variables son de valores múltiples o de valor continuo. Por lo tanto el método original de Chi-cuadrado debe ser extendido para trabajar con problemas más complejos. Para valores continuos de datos, el rango de valores debe ser discretizado en intervalos, antes de ser calculado el valor de Chi-cuadrado.

Sea N_{ij} el número de muestras de la clase C_i en el intervalo j, M_{ij} el número de muestras en el intervalo j y N el número total de muestras.

La frecuencia esperada de N_{ij} es $E_{ij} = M_{ij}|C_i|/N$.

La estadística de Chi-cuadrado puede definirse como:

$$CS = \sum_{i=1}^{C} \sum_{j=1}^{I} \frac{\left(N_{ij} - E_{ij}\right)^2}{E_{ij}} \tag{4.11}$$

Donde I es el número de intervalos. Mientras más grande sea el valor de CS mayor es la información que aportan las muestras de datos.

4.6. Escala Interactiva mejorada.

El método de escala interactiva mejorada (Improved Interactive Scaling IIS) [67] es un método desarrollado por IBM en el laboratorio T. J. Watson, es un algoritmo de subida de colina para calcular la máxima verosimilitud estimada de los parámetros del Modelo Condicional Exponencial (Condicional Exponencial Model ECM). El cual se expone a continuación:

Entrada: Un conjunto de T conexiones de muestra de una rede de entrenamiento con la función f_i

Sea B una función auxiliar

1. Inicializar todo $\lambda_{c,i}$ a 0.
2. Para $i = 1 : N$
3. Repetir
4. Calcular la clase esperada rotulada $p(c|x)$ para cada conexión de entrenamiento con los parámetros corrientes.
5. Resolver $\frac{\partial B}{\partial \delta_{c,j}} = 0$
   para cada $\delta_{c,j}$
6. Hacer $\lambda_{c,i} = \lambda_{c,i} + \delta_{c,j}$
7. Hasta que converja
8. Fin

Salida: Parámetro óptimo $\lambda_{c,i}$ y el modelo óptimo $p(c|x)$ resultando una clasificación que tome una conexión sin rotular y prediga su rotulo de clase.

La aplicación a la detección de intrusiones se realiza con el algoritmo anteriormente descripto teniendo en cuenta las siguientes premisas:

Una conexión de red solo pertenece a una sola clase (Normal o bajo ataque), para una conexión de prueba x con una N extraída de los descriptores de sucesos que maximice la probabilidad a posterior $p(c|x)$, la máxima verosimilitud para una instancia que tiene una clase rotulada c dada por la entrada x. Para el modelo condicional exponencial, esta probabilidad esta estimada por:

$$p(c|x) = \frac{1}{Z(x)} \exp\left(\sum_{i=1}^{N} \lambda_i f_i(x,c)\right) \tag{4.12}$$

Donde N es el número de sucesos . Y $f_i(x,c)$ es la función para el suceso i extraído de la entrada x y la salida c. λ_i es el valor real pesado correspondiente al suceso a ser estimado. El más alto peso, el suceso más significativo debería representar a la clase de interés. $Z(x)$ es el factor de normalización llamada función de partición, el cual asegura el rango de $p(c|x)$ entre 0 y 1 y fuerza a la suma $p(c|x)$ sobre los diferentes rótulos de clase a ser igual a 1 y es definida de la siguiente manera.

$$Z(x) = \sum_{c} \exp\left(\sum_{i=1}^{N} \lambda_i f_i(x,c)\right) \tag{4.13}$$

Para detección de anomalías en redes, cada clase usa el mismo conjunto de sucesos $f_i(x)$. Entonces la fórmula general puede ser reescrita de la siguiente forma:

$$p(c|x) = \frac{1}{Z(x)} \exp\left(\sum_{i=1}^{N} \lambda_{c,i} f_i(x)\right) \tag{4.14}$$

Donde $\lambda_{c,i}$ son los pesos c es el índice para la clase y i es el índice para el suceso. El proceso de entrenamiento para encontrar el conjunto de pesos que maximice la verosimilitud de los datos de entrenamiento Dada una distribución empírica d datos $p'(x,c)$ obtenida de los ejemplos de entrenamiento, la verosimilitud logarítmica L es la siguiente:

$$L = \sum_{x,c} p'(c|x) \log p(c|x) \tag{4.15}$$

$$= \sum_{x,c} p'(c|x) \sum_{i=1}^{N} \lambda_i f_{c,i}(x) - \sum_{c} p'(x) \log \left(\sum_{c} \exp \left(\sum_{i=1}^{N} \lambda_i f_{c,i}(x) \right) \right) \tag{4.16}$$

Donde $p'(x)$ es la distribución de los datos empíricos. Una aproximación iterativa para optimizar L es dividir el proceso de maximización en varios pasos o iteraciones. Los parámetros son actualizados en cada iteración para incrementar el logaritmo de verosimilitud de las iteraciones previas, y finalmente converger a un óptimo global, el cual es la solución de máxima verosimilitud para la forma paramétrica y la máxima entropía.

Si $\delta_{c,i}$ prmanece para diferencias en el peso de $\lambda_{c,i}$ entre dos iteraciones consecutivas, el cambio ΔL del logarítmo de verosimilitud será:

$$\Delta L = \sum_{x,c} p'(c|x) \sum_{i=1}^{N} \delta_{c,i} f_i(x) - \sum_{x} p'(x) \log \left(\sum_{c} p(c|x) \exp \left(\sum_{i=1}^{N} \delta_{c,i} f_i(x) \right) \right) \tag{4.17}$$

IIS usa la desigualdad $-\log \alpha \geq 1 - \alpha$ para desacoplar la interacción entre los parámetros $\{\delta_{c,i}\}$ debido a la función logarítmo y la desigualdad de Jensen:

$$\exp \left(\sum_{x} p(x) q(x) \right) \leq \sum_{x} p(x) \exp(q(x)) \tag{4.18}$$

ΔL puede ser limitado inferiormente con la función auxiliar B:

$$\Delta L \geq B = 1 + \sum_{x,c} p'(x,c) \sum_{i} \delta_{c,i} f_i(x) - \sum_{x} p'(x) \sum_{c} p(c|x) \sum_{i} \frac{f_i(x)}{f^{\#}(x)} \exp(\delta_{c,i} f^{\#}(x)) \tag{4.19}$$

Donde $f^{\#}(x)$ es la suma de todos los sucesos en el conjunto de entrenamiento. Se puede maximizar B en lugar de optimizar ΔL. Se encuentra el mejor $\{\delta_{c,i}\}$ diferenciando B y resolviendo por el máximo:

$$\frac{\partial x}{\partial \delta_{c,1}} = \sum_{x,c} p'(x) f_i(x,c) - \sum_{x} p'(x) \sum_{c} p(c|x) f_i(x,c) \exp(\delta_{c,i} f^{\#}(x,c)) \tag{4.20}$$

$\delta_{c,i}$ aparece sola, por lo tanto se puede resolver para cada parámetro en forma individual diferenciando B con respecto a cada $\delta_{c,i}$ en cada iteración.

4.7. Estadística de Kolmorov Smirnov

El procesador estadístico que se describe a continuación se utiliza para suministrar información a una red neuronal.

Debido a que el tráfico de red no es estacionario y los ataques a la red pueden tener diferentes duraciones (desde unos pocos segundos hasta varias horas), se necesita un procesador estadístico que sea capaz de supervisar eficientemente el tráfico de red con ventanas de observación de distinta duración. El uso de procesadores estadísticos para detección de intrusiones en redes ha sido ensayado por Nong [69]. En este trabajo se desarrolló un procesador estadístico tipo capa-ventana, mostrado en la Fig. 4.2, en donde cada capa-ventana corresponde a un lapso de tiempo supervisado con tamaño creciente. El esquema de la Fig. 4.2 se repite seis veces. Una vez para cada parámetro *i*. El procesador estadístico funciona de la siguiente forma:

Cada Función de densidad de probabilidad (Probability Density Funtion PDF) nueva, o reporte de evento, proveniente del procesador estadístico se almacena en el "buffer" de evento de la capa uno. Luego se actualiza el modelo de referencia. El evento almacenado se compara con el modelo de referencia de esa capa y el resultado Q (valor de similitud) se introduce en la red neuronal para decidir el estado de la red durante ese período de ventana. El "buffer" de eventos se vacía una vez que se llena y los eventos almacenados se promedian y se envían al "buffer" de eventos de la capa 2. Este proceso se repite en forma recursiva hasta alcanzar la última capa, donde los eventos son simplemente descartados después de ser procesados.

Los valores de similitud Q de la primera capa-ventana se actualizan cada 20 seg. los de la capa-ventana 2 cada 100 seg. los de la capa-ventana 3 cada 8 min. 20 seg y los de la capa-ventana 4 cada 41 min. 40 seg.

4.8. Algoritmos de actualización de los valores de similitud

En el sistema propuesto, la actividad de la red es muestreada y abstraída en PDF. Sin embargo, se usa una red neuronal para clasificación de ataques, que es capaz de aprender a partir de ejemplos de entrenamiento, para discernir degradaciones en la red, no interesa demasiado

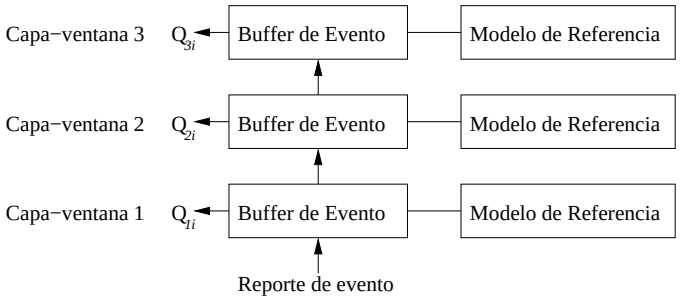

Figura 4.2: Diagrama de procesador estadístico

la distribución estadística de Q. A pesar de esto el algoritmo debe ser confiable al detectar las diferencias entre las PDF observadas y la referencia y adaptarse a los patrones de tráfico observados, ya que el tráfico de red no es estacionario y a que los ataques de red pueden tener diferentes duraciones. El algoritmo de medición de similitud es el propuesto por Manikopoulos [13] que se muestra a continuación:

$$Q = f(N) \cdot \left[\sum_{i=1}^{k} \left| p_i' - p_i \right| + max_{i=1}^{k} \left(\left| p_i' - p_i \right| \right) \right] \tag{4.21}$$

Donde $f(N)$ es una función del número de muestras PDF. El valor de similitud Q se lo puede expresar en función de N y Q' para ello se utiliza la función $Q = f(Q', N)$ donde:

$$Q' = \left[\sum_{i=1}^{k} \left| p_i' - p_i \right| + max_{i=1}^{k} \left(\left| p_i' - p_i \right| \right) \right] \tag{4.22}$$

Sea S el espacio de muestras de una variable aleatoria y los eventos $E_1, E_2, ...E_k$ una partición mutuamente exclusiva de S. Se supone que p_i es la probabilidad de ocurrencia esperada del evento E_i, que corresponde al modelo de referencia y p_i' es la frecuencia de ocurrencia del evento E_i durante un período de tiempo, que corresponde a las muestras observadas. El valor Q' varia entre 0 y $3.N$ donde N es el número de muestras PDF. Para facilitar el entrenamiento de la red, se necesita acotar Q de manera que:

$$-1 \leq Q \leq 1 \tag{4.23}$$

para ello se utiliza la siguiente función:

$$Q\left(Q',N\right) = -\frac{2.Q'}{3.N} + 1 \tag{4.24}$$

Así se logra que Q sea una variable continua entre -1 y 1. Aparte de la medición de similitud también se usa un algoritmo para la actualización en tiempo real del modelo de referencia. Sea p_{old} el modelo de referencia antes de la actualización, p_{new} el modelo de referencia posterior a la actualización, p_{obs} la actividad observada dentro de una ventana de tiempo. La fórmula para actualizar el modelo de referencia es:

$$p_{new} = s.\alpha.\overline{p}_{obs} + (1 - s.\alpha).\overline{p}_{old} \tag{4.25}$$

en donde:

α es el coeficiente de aprendizaje.

s es la función de adaptación dinámica.

Se supone que la salida de la red neuronal es una variable continua u entre -1 y 1, donde -1 significa falla con absoluta certeza y 1 significa normalidad con plena seguridad. Los valores intermedios de u indican niveles de certeza proporcionales. La función para calcular s es:

$$s = \begin{cases} u & \text{si } u \geq 0 \\ 0 & \text{si } u < 0 \end{cases} \tag{4.26}$$

De esta forma, se asegura una rápida actualización para tráfico normal, mientras que permanece sin cambios cuando ocurren ataques.

Capítulo 5

Redes Neuronales

5.1. Introducción

Muchos investigadores coinciden en definir las redes neuronales como una red que de elementos procesadores simples (neuronas), los cuales pueden exhibir un comportamiento global complejo, determinado por por las conexiones entre ellos y alguno elementos de parámetros. La inspiración original para está técnica resulto del examen del sistema nervioso central y las neuronas con sus axones, dendritas y sinapsis que constituye uno de los más significativos elementos de procesamiento. En el modelo de red neuronal, nodos simples llamados neuronas, neuronodos, elementos de procesamiento, o unidades, son conectadas entre ellos para formar una red de nodos, de aquí el nombre de red neuronal. Mientras una red neuronal no tiene que ser necesariamente adaptativa, se usa con un algoritmo diseñado para alterar los pesos o sea la fuerza de interconexión entre ellos en la red para producir el flujo deseado de información.

Estas redes son similares a las redes neuronales biológicas en el sentido de que las funciones son realizadas en forma colectiva y en paralelo por cada unidad, más que estableciendo una delimitación de subtareas a las cuales varias unidades son asignadas. Corrientemente el término Redes neuronales artificiales tiende a referirse principalmente a los modelos de redes neuronales usados en estadísticas, sicología cognoscitiva y inteligencia artificial.

En las implementaciones de redes neuronales artificiales la aproximación basada en la biología ha sido más o menos abandonada por una aproximación más practica basada en las estadísticas y el procesamiento digital de señales. En algunos de estos sistemas de redes neuronales, como parte de las redes neuronales son usados como partes de componentes de grandes sistemas que combinan elementos adaptativos y no adaptativos. Mientras que una aproximación de tales sistemas adaptativos es más pertinente para solucionar problemas de la vida real, esto esta más lejos que los modelos conectivistas de la inteligencia artificial tradicional. Que sin embargo tienen en común los principios de no linealidad, distribuidos, paralelos y procesamiento local adaptativo. [70].

5.2. Modelos

Los modelos de redes neuronales usados en Inteligencia Artificial son usualmente simples modelos matemáticos definidos por la función:

$$f : X \to Y \tag{5.1}$$

La palabra red en los términos de redes neuronales artificiales surge porque la función $f(x)$ está definida por la composición de otras funciones $g_i(x)$, las cuales a su vez pueden ser definidas como una composición de otras funciones. Esto puede ser convenientemente representado como una estructura de red, con flechas representando las dependencia entre las variables. Un amplio tipo de composición de redes neuronales artificiales es la de la suma no lineal pesada donde:

$$f(x) = K\left(\sum_i w_i g_i(x)\right) \tag{5.2}$$

Donde K es una función predefinida como tangente hiperbólica ecuación 5.8. De aquí en adelante se referirá g_i como el vector $g = (g_1, g_2, g_3, ...g_n)$

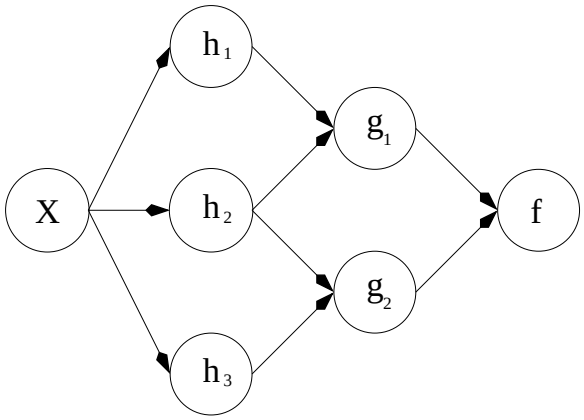

Figura 5.1: Gráfico de dependencia en una red neuronal artificial

La figura 5.1 muestra la función f descompuesta entre variables cuyas dependencias están indicadas por las flechas, esto se puede interpretar de dos formas:

- Una vista es la funcional: La entrada x es transformada en un vector tridimensional h el cual es transformado a su vez en un vector bidimensional g el cual es finalmente transformado en f. Este modo es frecuentemente encontrado en el contexto de optimización.

- Una segunda vista es probabilística :La variable aleatoria $F = f(G)$ depende a su vez de la variable aleatoria $G = g(H)$ la cual a su vez depende de la variable aleatoria $H = h(X)$ la que a su vez finalmente depende de la variable X. Esta vista es frecuentemente utilizada en los modelos gráficos.

Ambas vistas son equivalentes y los componentes de cada capa son independientes entre si lo que permite un alto grado de paralelismo en las implementaciones. Redes neuronales como las descriptas en la figura 5.1 son llamada de alimentación directa, porque el gráfico que la representa no tiene ciclos cerrados. Redes con ciclos como muestra la parte superior de la figura 5.2 son llamadas recurrentes en la cual f es dependiente de ella misma. Sin embargo a pesar de que no se muestra en la figura el valor de f depende en algún punto del tiempo t y de tiempos anteriores. En la parte inferior de la figura 5.2 ilustra el caso donde la la variable f depende de t y de su ultimo valor.

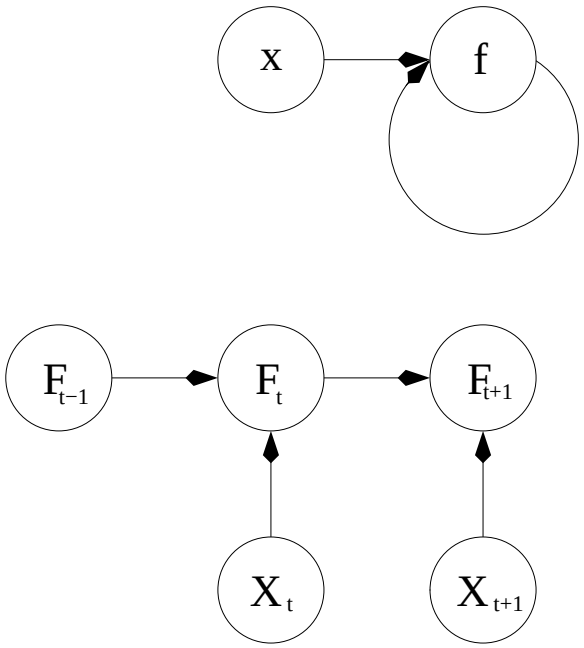

Figura 5.2: Gráfico de dependencia recursiva en una red neuronal artificial

5.3. Aprendizaje

Uno de los elementos más atractivos de las redes neuronales es su capacidad de aprendizaje, lo que en términos matemáticos significa lo siguiente: Dada una tarea especifica para resolver, y una función de clase F, aprendizaje significa que usando un conjunto de observaciones con el objetivo de encontrar una función $f^* \in F$ la cual resuelva la tarea en la forma más eficiente. Esto lleva a definir una función cuyo costo $C : F \to \mathbb{R}$ tal que para una solución óptima $f*, C(f^*) \leq C(f) \forall f \in F$ (ninguna solución tiene un costo menor que la solución óptima).

La función costo C es un concepto importante, dado que es una medida de cuan lejos nos encontramos de la solución óptima al problema que se quiere resolver. Los algoritmos de aprendizaje buscan a través del espacio de soluciones en orden a encontrar una función que tenga el menor costo posible.

Para aplicaciones donde la solución depende de los datos, el costo debe ser necesariamente una función de las observaciones, de otro forma no se estaría modelando algo relacionado con los datos. Esto es algo frecuentemente definido como estadístico para lo cual solo se pueden realizar aproximaciones. Como un simple ejemplo se puede considerar el problema de encon-

trar f la cual minimice $C = E\left[(f(x) - y)^2\right]$, f para el par de datos (x, y) arrojados desde una distribución $\mathcal{D}$. En situaciones prácticas solo se pueden obtener N muestras de $\mathcal{D}$ y para este ejemplo solo se debe minimizar $\hat{C} = \frac{1}{N}\sum_{i=1}^{N}(f(x_i) - y_i)^2$. Tal que el costo es minimizado sobre la muestra de datos y no sobre la distribución verdadera de estos.

Cuando $N \to \infty$ alguna forma de aprendizaje en linea sera usado, cuando el costo es parcialmente minimizado cuando cada nueva muestra es vista. Mientras el aprendizaje en linea es a menudo usado cuando $\mathcal{D}$ es fijo, y es más usado cuando la distribución cambia muy lentamente en el tiempo. En redes neuronales el aprendizaje en linea es frecuentemente usado cuando el conjunto de datos es finito.

5.4. Paradigmas de aprendizaje

Existen tres principales paradigmas de aprendizaje, cada uno corresponde a una tarea abstracta de aprendizaje las cuales son:

- Aprendizaje supervisado.
- Aprendizaje no supervisado.
- Aprendizaje reforzado.

5.4.1. Aprendizaje supervisado.

En el aprendizaje supervisado, se toma un conjunto por ejemplo los pares $(x, y), x \in X, y \in Y$ con el objetivo de encontrar una función $f : X \to Y$ dentro de la clase de funciones permitidas que mejor aparee el ejemplo.
En otra palabras, se desea inferir la aplicación implicada en los datos; el costo de la función está relacionada con la falta de apareamiento entre los datos del conjunto de salida y el de llegada y implícitamente contiene conocimientos a priori acerca del dominio del problema.

Una función de costo comúnmente usada es el error medio cuadrado el cual trata de minimizar el error entre la salida de la red, $f(x)$ y el valor de destino sobre todos los pares de ejemplos. Cuando se trata de minimizar este costo usando gradientes descendentes para las redes neuronales llamadas perceptron multi capa, se obtiene el bien conocido algoritmo de entrenamiento llamado "back propagation". Tareas que entran en el aprendizaje supervisado son reconocimiento de patrones, también conocido como clasificación y regresión conocido como función aproximación este tipo de aprendizaje es aplicable a datos secuenciales y es el más apto para IDS.

5.4.2. Aprendizaje no supervisado.

En el aprendizaje no supervisado dado un dato x y una función costo a ser minimizada, puede ser cualquier función del dato x y de la salida de la red neuronal f. La función costo es dependiente de la tarea que se este tratando de modelizar y de las asunciones a priori propiedad implícita del modelo, sus parámetros y las variables observadas.

Como ejemplo trivial considérese el modelo $f(x) = a$, donde a es una constante y la función costo $C = E[(x - f(x))2]$. Minimizando este costo se obtendrá un valor que es igual al valor medio de los datos. La función costo puede ser mucho más complicada y su forma depende de la aplicación. Por ejemplo en compresión puede estar relacionada por la información mutua entre x e y. En modelado estadístico, puede estar relacionada con la probabilidad a posterior del modelo dado de datos, cabe destacar que en ambos ejemplos las cantidades deben ser maximizadas y no minimizadas.
Las tareas cubiertas por este paradigma son en general problemas de estimación, distribuciones estadísticas, compresión y filtrado.

5.4.3. Aprendizaje reforzado.

En aprendizaje reforzado, el dato x usualmente no es suministrado, sino generado por un agente por interacciones con el ambiente. En cada tiempo t el agente realza una acción yt y el ambiente genera una observación xt y un costo instantáneo ct, de acuerdo a alguna dinámica conocida. El objetivo es generar una política para seleccionar las acciones que minimice a largo término el costo ct. Por ejemplo el costo esperado acumulativo. la dinámica de ambiente y el costo a largo término para cada política es usualmente desconocido pero puede ser estimado.
El ambiente es modelado como un proceso de decisión de markov (MDP) con los estados $s_1, ..., s_n \in S$ y las acciones $a_1, ..., a_m \in A$ con la siguiente función de distribución de probabilidad: la distribución instantánea de costo $P(ct|st)$, the observation distribution $P(xt|st)$ y la transición $P(st+1|st, at)$, mientras que la política es definida como la distribución condicional sobre las acciones de las observaciones. Tomadas en conjunto, las dos definen una cadena de Markov (MC). El objetivo es descubrir la política que minimice el costo.
Las tareas cubiertas por este paradigma son los problemas de control juego y otras decisiones secuenciales.

5.5. Algoritmos de aprendizaje

Entrenamiento de una red neuronal esencialmente significa seleccionar un modelo de un conjunto de modelos permitidos, que minimice el criterio de costo. Existen numerosos algoritmos disponibles para entrenamiento de redes neuronales, algunos de ellos pueden ser vistos como aplicaciones directas de la teoría de optimización y estimación estadística.
La mayoría de los algoritmos usados en el entrenamiento de redes neuronales artificiales consiste en el empleo del gradiente descendente. Esto se realiza tomando la derivada de la función costo con respecto a los parámetros de la red y cambiando esos parámetros en la dirección relativa al gradiente.
Métodos evolucionarios de aprendizaje simulan la maximización de la esperanza y de métodos no paramétricos. se basan en encontrar relaciones temporales en las cadenas de la señales de

los sensores.

Métodos de aprendizaje de percepción se basan en encontrar relaciones temporales en las cadenas de la señales de los sensores. En un ambiente estadísticamente saliente de las correlaciones temporales, que pueden encontrarse monitorizando el tiempo de arribo de las señales de los sensores.

5.5.1. Función de activación.

En una red neuronal "perceptron" múltiple capa la función de activación lineal consiste en un mecanismo "abierto cerrado" para determinar cuando la neurona se activa, entonces es fácil comprobar mediante una simple demostración de álgebra lineal que una red de múltiples capas puede ser reducida a un modelo de dos capas de entrada y salida. Lo que la hace a la red neuronal perceptron de múltiples capas es que cada neurona usa una función de activación no lineal la que es desarrollada para modelar la acción de potencial de activación de las neuronas biológicas. está función es modelada en varias formas, pero siempre debe ser normalizable y diferenciable.

Las principales funciones de activación corrientemente usadas son sigmoides que son descriptas a continuación

$$\phi(v_i) = \tanh(v_i) \text{ and } \phi(v_i) = (1 + e^{-v_i})^{-1}, \tag{5.3}$$

De las cuales una de ellas es la tangente hiperbólica que se usara más adelante y cuyo valor está en el rango $-1, 1$, y la segunda es equivalente en la forma pero su rango está entre $0, 1$. Donde y_i es la salida del nodo i-ésimo (neurona) y v_i es el peso de la suma de las entradas sinápticas. Funciones de activación más especializadas incluyen base radial que son usadas por otras clases de modelos de redes neuronales .

5.5.2. Capas

El perceptron multicapa consiste de una entrada y una salida con una o más capas ocultas con funciones de activación las neuronas no lineal y una capa se conecta con un cierto peso w_{ij} a cada otro nodo de la siguiente capa.

5.5.3. Aprendizaje a través de Backpropagation

El aprendizaje ocurre en la perceptron cambiando los pesos de las conexiones después de que cada dato ha sido procesado, basado en el valor del error a la salida comparado con el valor esperado. Este es un ejemplo de aprendizaje supervisado y es llevado a cabo a través de "backpropagation", una generalización del algoritmo de los cuadrados mínimos en el perceptron lineal.

Se representa el error en el nodo de salida j en el n-ésimo nodo de dato por $ej(n) = dj(n) - yj(n)$, donde d es el valor esperado y y es el valor producido por la red "perceptron". Entonces se realizan correcciones en los pesos de los nodos basados esas correcciones que minimizan la energía del error en toda la salida dada por la siguiente ecuación:

$$\mathcal{E}(n) = \frac{1}{2}\sum_{j} e_j^2(n). \tag{5.4}$$

Por la teoría de los diferenciales, se puede encontrar que el cambio en cada nodo debe ser:

$$\Delta w_{ji}(n) = -\eta \frac{\partial \mathcal{E}(n)}{\partial v_j(n)} y_i(n) \tag{5.5}$$

Donde *yi* es la salida de la neurona previa y η es la velocidad de aprendizaje, la cual es cuidadosamente elegida para asegurar que los pesos converjan a una repuesta que no sea ni muy especifica ni muy general. En la generalidad de la aplicaciones típicamente el rango está entre los siguientes valores $0{,}2, 0{,}8$.

La derivada puede ser calculada dependiendo de la suma v_j de la entrada sináptica. Esto es fácil

de probar que para un nodo de salida la derivada puede ser simplificada a:

$$-\frac{\partial \mathcal{E}(n)}{\partial v_j(n)} = e_j(n)\phi'(v_j(n)) \tag{5.6}$$

Donde ϕ' es la derivada de la función de activación descripta en la parte superior, la cual por si sola no varia. El análisis es más dificultoso por los cambios de pesos en los nodos ocultos, pero puede ser demostrado que la derivada relevante es:

$$-\frac{\partial \mathcal{E}(n)}{\partial v_j(n)} = \phi'(v_j(n)) \sum_k -\frac{\partial \mathcal{E}(n)}{\partial v_k(n)} w_{kj}(n). \tag{5.7}$$

Notar que está depende de los cambios en los pesos de los nodos k-ésimo, la cual representa la capa de salida. De tal forma que para cambiar los pesos en las capas ocultas, se debe primero cambiar los pesos de la capa de salida de acuerdo a la derivada de la función activación, y este algoritmo representa la "backpropagation" de la función activación.

5.6. Implementación de la red neuronal.

La salida del procesador estadístico es un vector ordenado k dimensional, cuyos componentes son los valores de similitud entre las PDF medidas y las de referencia de los k parámetros de red adquiridos, durante una ventana de observación. En este caso $k = 6$ y se tienen tres valores de similitud por cada parámetro. Este vector k-dimensional representa una evaluación del estado de la red o del sistema durante el período de observación. A continuación este vector de estado se presenta al clasificador multivariable para obtener un resultado de clasificación. El vector ordenado k dimensional generado por el procesador estadístico para el clasificador, puede ser pensado como un patrón de entrada. Las redes neuronales se consideran como una aproximación efectiva en el manejo y clasificación de patrones. Existen varios tipos de redes neuronales como las clasifica Bishop [71] tales como la "perceptrón simple", "back propagation", "perceptron-back propagation hybrid", "Radial-Basis Function", "Fuzzy ART MAP", entre otras.

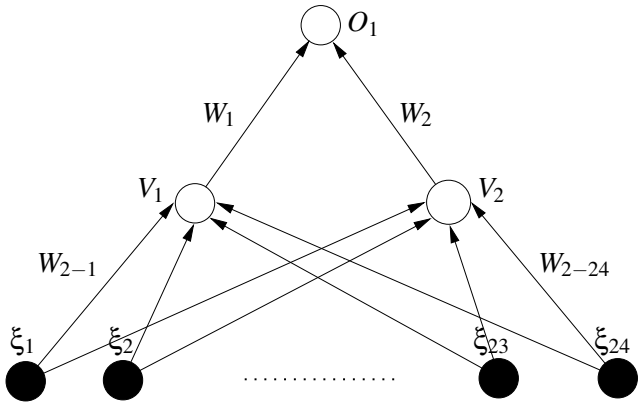

Figura 5.3: Diagrama de la red neuronal

Se seleccionó e implementó la "back-propagation" por poseer mejor rendimiento de clasificación en este tipo de problemas. Se puede ver una aplicación similar en el trabajo de Ramasubramanian [72] y en el trabajo de Lee [73]. Con 6 parámetros de red a supervisar y 3 valores de similitud por parámetro, se obtiene un total de 18 valores de similitud. Por lo tanto se necesita una red neuronal con 18 entradas. En la Fig. 5.3, se utilizan dos neuronas en la capa oculta. Esto se debe a que al aumentar el número de neuronas se mantiene constante el error de entrenamiento. En consecuencia, para disminuir el tiempo de procesamiento, se adopta la mínima cantidad de neuronas en la capa oculta.

Para todas las neuronas, tanto las de la capa oculta como la de salida, se utiliza la función tangente hiperbólica como función de activación $g(h)$:

$$g(h) = \tanh(\beta h) \tag{5.8}$$

El parámetro β es la pendiente de la función $g(h)$ para $h = 0$. Se utiliza un valor de $\beta = 1,8$ para la neurona de la capa de salida y un valor de $\beta = 0,9$ para las neuronas de la capa oculta. Estos valores fueron obtenidos luego de realizar ajustes en ambos parámetros a los fines de minimizar el error de entrenamiento y de generalización de la red. Como puede observarse en la Fig. 5.4, la pendiente de la neurona de la capa de salida ($\beta = 1,8$) es mayor que las de la neuronas de

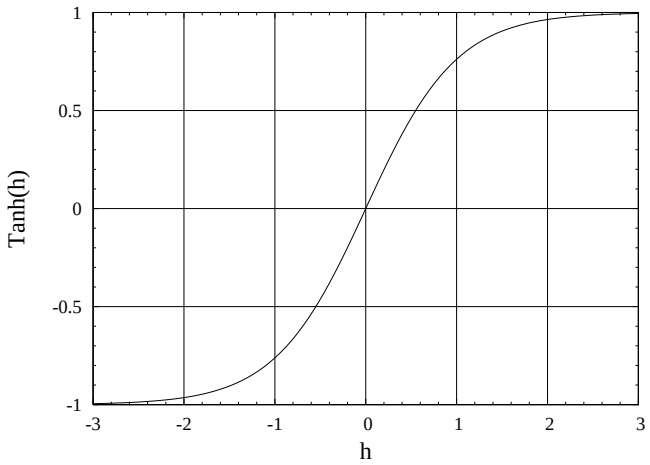

Figura 5.4: Función tangente hiperbólica

la capa oculta ($\beta = 0{,}9$), para facilitar la saturación de la función de activación en los valores limites ± 1.

Para mejorar el algoritmo de actualización de los pesos, se agregó un término de momento como se sugiere en el trabajo de Allred [74] de acuerdo a la ecuación:

$$\Delta W_{pq}(t+1) = -\eta \frac{\partial E}{\partial W_{pq}} + \alpha \Delta W_{pq}(t) \tag{5.9}$$

Se estableció el parámetro de momento, en el valor $\alpha = 0,5$ y la velocidad de aprendizaje $\eta = 0,2$ después de probar con varios valores y alcanzar el mínimo error de la red neuronal.

5.7. Entrenamiento de la red neuronal.

Una vez desarrollado el algoritmo de la red neuronal y todas las etapas anteriores, resta realizar el entrenamiento de la red neuronal. Es necesario tener en cuenta que de un buen entrenamiento depende el correcto desempeño de todo el sistema. Para llevar a cabo dicho entrenamiento se usó el modo incremental, que consiste en presentar un patrón a la entrada y luego actualizar los pesos antes de presentar el patrón siguiente. El modo incremental se desarrolla mediante el siguiente algoritmo:

1. Inicializar los pesos a valores aleatorios pequeños.

2. Elegir un patrón y aplicarlo a la entrada de la red.

3. Calcular las salidas de todas las neuronas.

4. Calcular los deltas para la capa de salida, comparando la salida real con la salida deseada.

5. Calcular los deltas para la capa oculta.

6. Actualizar los pesos con la incorporación del término de momento.

7. Volver al paso 2 y repetir el proceso hasta llegar al último patrón.

8. Repetir desde el paso 2 un número definido de iteraciones. (En este caso 150 veces).

9. Verificar que el error de aprendizaje y generalización sean aceptables. De lo contrario ajustar los parámetros de la red (α de la red neuronal, α del procesador estadístico, β y η) y repetir todo el proceso nuevamente.

5.8. Obtención de los parámetros de entrada y salida.

Los patrones utilizados en el entrenamiento constituyen un conjunto de entradas/salidas conocidas. Este conjunto está integrado tanto de patrones de tráfico normal de la red como de patrones de tráfico bajo condición de ataque. Para obtener los patrones de entrada de la red neuronal (vector k-dimensional cuyos componentes son los valores de similitud) se utiliza el programa PCAP31 que incluye solamente el bloque de captura. El conjunto de muestras de entrenamiento es obtenido en un servidor ("Firewall" de la UNC). Se adquirieron un total de 360 muestras de las cuales 180 corresponden a tráfico de red normal y las restantes a tráfico de ataque, el que resulta de superponer al tráfico normal, el tráfico producido por un programa generador de ataque. Además se obtuvieron otras 360 muestras para verificar la generalización de la red neuronal. La salidas del programa PCAP31 se almacenaron en un archivo, para luego ser utilizadas en el cálculo de los valores de similitud. Una vez obtenido el archivo con las muestras de entrada, se procedió al entrenamiento de la red propiamente dicha. Para ello se

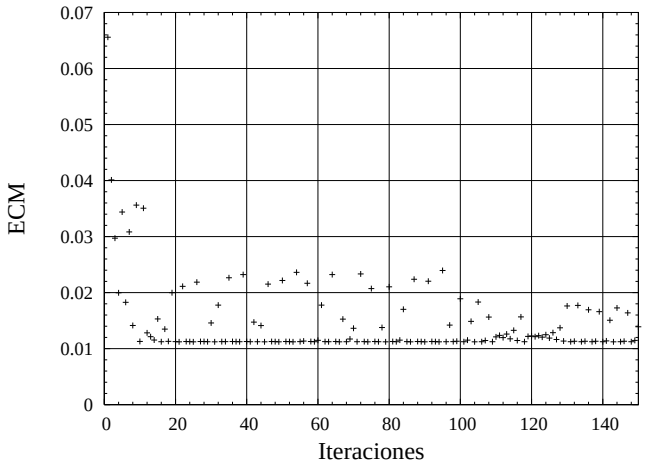

Figura 5.5: Error cuadrático medio

utilizó el programa TRAINING3. Este programa lee los datos del archivo de las muestras de la red y las procesa estadísticamente, generando los patrones de entrada para la red neuronal. Los patrones de salida, son conocidos, mediante el registro de los períodos de tiempo en los cuales se realizaron los ataques. Se asignó 1 para condición de tráfico normal y -1 para la condición de ataque. Una vez obtenidos los patrones de entrada y salida, se realizó el entrenamiento de la red con el programa TRAINING3, que sigue los pasos descriptos en la sección X. Los pesos calculados de la red se almacenan en un archivo, que se utilizará en el próximo paso. El error de aprendizaje resultante fue de $0,56\,\%$. Este error es el porcentaje de salidas de la red neuronal con falsos positivos y falsos negativos en relación al total de salidas. En la Fig. 5.5 se observa la convergencia del error cuadrático medio de la red neuronal, a medida que transcurren las iteraciones. Es importante notar que el error cuadrático medio disminuye muy rápidamente luego de sólo algunas iteraciones, alcanzando niveles de convergencia satisfactorios dentro de las 15 iteraciones. Estas características son especialmente deseables para sistemas de detección de intrusiones en redes (NID), los cuales necesitan supervisión en tiempo real y entrenamiento en línea. Una vez realizado el entrenamiento, se utiliza el segundo conjunto de 360 muestras de tráfico de red y el programa GENERALIZAR para verificar el correcto aprendizaje de la red neuronal y analizar su generalización. El error de generalización obtenido fue del $1,97\,\%$.

En primer lugar, se comprueba que la red neuronal puede entrenarse fácilmente para distinguir entre condiciones normales y en condiciones de ataque de la red. Se obtuvo un error

de aprendizaje de $0,56\,\%$ y un error de generalización de $1,97\,\%$, utilizando una configuración sencilla de la red neuronal (configuración "back-propagation" con sólo dos neuronas en la capa oculta). Este resultado se ha logrado por el uso del módulo procesador estadístico que colabora en forma eficiente en la detección de intrusiones.

Como se puede advertir, para la detección de inundaciones UDP, el aporte realizado por los parámetros relacionados con el protocolo IP es pequeño en relación al aporte de los parámetros del protocolo UDP al momento de llevar a cabo la detección. Por esta razón, se podría omitir la utilización de los parámetros del protocolo IP.

El sistema se podría adaptar para detectar otros tipos de intrusiones, tales como inundaciones TCP-SYN e ICMP (Internet Control Message Protocol, Protocolo Internet de Mensajes de Control).

En el caso de inundaciones TCP-SYN, se debería incluir el uso de parámetros relacionados al protocolo TCP mientras que las inundaciones de paquetes ICMP se detectarían a través de los parámetros del protocolo IP. Esto será factible siempre que la red neuronal pueda aprender de los distintos tipos de ataque.

Se debe tener en cuenta que en este caso, no se tiene control sobre el tráfico de fondo de la red. Por el contrario si se tuviese contro de este, a través de programas de simulación y de modelado de tráfico de red, se podría analizar el rendimiento del sistema para distintos tráficos de fondo y distintos niveles de ataque para mejorar su desempeño.

Con el sistema IDS implementado, se detectaron ataques en la red, se dejó para una etapa posterior elaborar acciones defensivas a través de la construcción automática de reglas al "firewall".

Capítulo 6

DPI

6.1. Inspección Profunda de Paquetes

6.1.1. Introducción

Los algoritmos usados por décadas para comparación de cadenas de caracteres, recientemente se ha probado que son útiles para para la inspección profunda de paquetes TCP/IP [75], para detectar intrusiones, buscar virus y filtrar contenidos de Internet. Si embargo los algoritmos deben ser optimizados, para volverse eficientes para procesar flujos de datos a velocidades de multigigabits y manejar un gran volumen de firmas.

Antes del 2001 las investigaciones en procesamiento de paquetes estaban orientadas a búsquedas del prefijo más largo (longest prefix matching) en una tabla de ruteo se busca la red con mascara de subred más grande y clasificación de paquetes por varios campos del encabezamiento (Multifield packet classification) en el caso de firewalls y aplicaciones de calidad de servicio. En la detección de intrusiones en redes, búsqueda de virus, filtrado de contenidos, administración de mensajería instantánea y sistemas peer to peer, se trabaja con búsqueda de patrones en cadenas de caracteres para realizar la inspección de los datos. Se han hecho grandes avances en el diseño de algoritmos como en las implementaciones de hardware para acelerar la inspección, reducir las necesidades de almacenamiento y manejar en forma más eficiente las expresiones regulares.

6.2. Desarrollo de algoritmos de búsqueda en cadena de caracteres

Técnicas de paralelismo de Bits se utilizan con frecuencia en biología computacional, pero rara vez en la inspección de redes. Asumimos que la longitud del texto para ser inspeccionado es de *n* caracteres y la longitud del patrón a buscar o (la longitud más corta en el caso de varios patrones) es de *m* caracteres.

6.2.1. Enfoque basado en Autómatas

Un enfoque basado en autómata busca coincidencias de patrones parciales, ya sea en una transición de estados de autómata finito determinista(DFA Deterministic finite automaton) o de un autómata finito no determinista (NFA Nondeterministic finite Automaton). Una aplicación general DFA tiene menor complejidad pero exige más espacio para el almacenamiento de patrones, mientras que una aplicación NFA es todo lo contrario.

El enfoque basado en autómata es popular en DPI por dos razones:

- La ejecución en un tiempo finito determinado garantiza el peor de los casos, aun ante presencia de ataques diseñados deliberadamente para explotar un algoritmo.
- Construir un autómata que acepte expresiones regulares es algo sistemático y bien estudiado.

Dada la capacidad de las lineas de datos de las arquitecturas de la computadoras modernas de 32 y 64 bits, la búsqueda en autómatas con una entrada de un carácter, resulta en una muy pobre utilización de las arquitecturas modernas y degrada el rendimiento. La ampliación de la tabla de transición para guardar las transiciones de dos o más caracteres es plausible, pero es impracticable sin la debida compresión de las tablas. Almacenar un gran conjunto de patrones pre establecido también consume importantes recursos e memoria debido al gran número de estados. Investigaciones recientes, tratan de reducir la estructura de datos espaciales y al

mismo tiempo inspeccionar múltiples caracteres. Una estructura de datos compactos en la implementación del programa incrementa el rendimiento debido a un mejor aprovechamiento del cache.

6.2.1.1. Reducción de las tablas de transición extensas

La reducción de tablas transición extensas, en general los nodos alejados de la raíz tienen pocos estados próximos validos.

Podemos comprimir la tabla almacenando sólo enlaces a estados próximos validos después de uno o más caracteres de entrada y enlaces fallidos de cada estado. También se puede almacenar la tabla de estado de transición de vínculos fallidos y la lista de estados finales de búsqueda exitosa en forma separada en los programas de aplicación a los fines de aprovechar mejor la localidad del cache. Snort (www.snort.org), ha optimizado cuidadosamente las estructuras de datos de está forma mejorando el rendimiento de caché. La última revisión utiliza una construcción básica NFA como método de búsqueda por defecto. Se puede ver la implementación de Snort (en /sfutil/bnfa_search.c en el árbol de las fuentes de Snort 2.6.1).

6.2.1.2. Reducción de transiciones

La reducción de las transiciones. Con el alfabeto ASCII extendido, un autómata tiene un máximo de 256 transiciones de estado. La división de un autómata en varios más pequeños a nivel de bit puede reducir el número de transiciones. Por ejemplo, supongamos el autómata se divide en ocho y, a continuación, un autómata es alimentado con el bit “b7” el siguiente con el bit “b6” y así sucesivamente, donde b7,b6,b5,b4,b3,b2,b1,b0 son los ocho bits de un carácter de entrada. Este método puede ser implementado en hardware para seguir ese autómata en paralelo. Estos autómatas son compactos porque cada entrada tiene a lo máximo dos transiciones de estado validas para cada bit de entrado 0 o 1. Expandiendo el autómata para leer múltiples entradas simultáneamente es facilitado debido a la reducción de los datos de salida. En este caso solo hay 16 transiciones de estado validas para cuatro caracteres de entrada. Dado que el grupos de estados en un autómata generalmente tienen transiciones de salidas comunes que conducen

al mismo estado para el mismo grupo de caracteres de entrada el método de entrada retardada (D^2FA Delayed input Finite Automaton) puede reducir el número de transiciones. Un estado en un grupo puede mantener solo sus transiciones únicas y hacer la transición default para el estado en el grupo responsables de las transiciones comunes. Este método es responsable de la reducción en un 95 % de las transiciones en aplicaciones prácticas.

6.2.1.3. Tablas de Hash

Una tabla hash puede almacenar las transiciones desde un estado inicial de un autómata hasta su correspondiente próximo estado válido después de algunos caracteres de entrada. Realizando el seguimiento de múltiples caracteres de entrada al mismo tiempo convirtiéndose en una tabla de consulta (Lookup-table). A raíz de que solos unos pocos caracteres de entrada pueden conducir a los próximos estados válidos, las tablas hash pueden ser fácilmente manejables. Un filtrado adicional puede lograr búsquedas exitosas en las tablas hash para acelerar este método.

6.2.1.4. Agrupación y reescritura

Algunas combinaciones de comodines y repeticiones pueden generar autómatas muy complejos que crecen exponencialmente. Es posible reescribir las expresiones regulares para simplificar el autómata debido que no se debe encontrar cada aparición de la firma buscada en la cadena de entrada del flujo de datos de la red, solo basta identificar la primera aparición. Por ejemplo cada cadena *s* identificada por “ab+” (el + en las expresiones regulares denota uno o mas) puede ser identificado por “ab” como *s* o el prefijo *s*, por lo tanto reportando una coincidencia con “ab” es suficiente. Además compilando toda la expresión regular en un solo autómata, puede resultar un autómata muy complejo. En un ambiente multiprocesador como es común hoy en día, se puede agrupar expresiones regulares en autómatas separados de acuerdo a la interacción entre ellos.

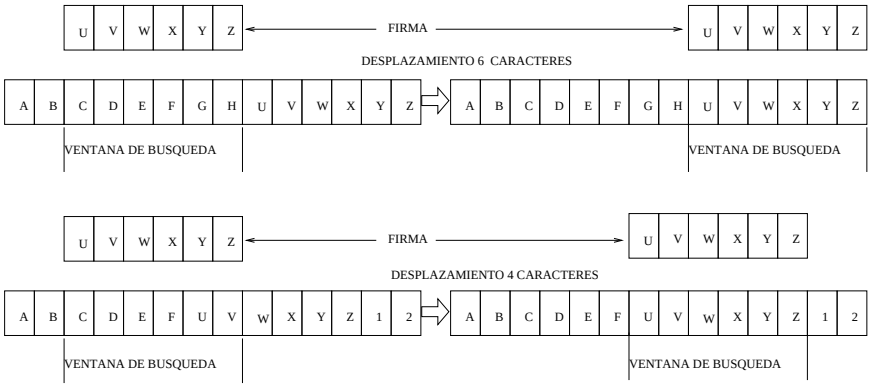

Figura 6.1: Ejemplo de aproximación heurística

6.2.2. Aproximaciones basadas en heurística

Una aproximación basada en heurística puede pasar por alto ciertos caracteres que no coincidan para acelerar la búsqueda basada en cierta heurística. Durante la búsqueda, una ventana de búsqueda de m caracteres cubre el texto bajo inspección y es deslizada a través del texto. Una búsqueda heurística puede buscar las coincidencias dentro de la ventana por una apariencia de la firma buscada. Esta determina donde existe una coincidencia sospechosa y se mueve a la próxima ventana si no sucede así. A causa de que las posiciones o los desplazamientos a posibles bloques con computados y almacenados en tablas de antemano, la tabla de búsqueda maneja los desplazamientos de la ventana de búsqueda.

En la figura 6.1 se muestra una búsqueda basada en heurística muy simple para solo una firma, para visualizar como se puede saltar en forma eficiente. En la parte superior de la figura 6.1 “FGH” no es una subcadena de la firma a buscar y el sufijo no es un prefijo de la firma, por lo tanto se desplaza la ventana de búsqueda por $m = 6$ caracteres sin examinar el resto de los caracteres de la ventana ya que no podrá haber una coincidencia. Luego del desplazamiento “XYZ” es un sufijo tanto de la cadena como de la firma, por lo tanto la ventana entera es verificada, encontrándose la coincidencia. Sin embargo si el sufijo de la ventana de la cadena es igual al prefijo de la firma, el valor a desplazar la ventana más, podría ser menor que m porque el sufijo puede ser el prefijo de la firma después del desplazamiento como muestra la parte inferior de la figura 6.1. Se puede extender está heurística para manejar firmas más cortas que que el tamaño de bloque. Si la firma es una subcadena del bloque, mirando el bloque se puede re-

clamar una coincidencia. En adición a la búsqueda de coincidencias heurísticas, para encontrar coincidencias en cadenas fijas, Gonzalo Navarro y Mathieu Raffinot [76] presentan un método heurístico para saltar un texto de caracteres para coincidencias en expresiones regulares.

6.2.3. Aproximación basada en Filtrado

La aproximación basada en filtrado busca texto con determinadas características y rápidamente excluye aquellas cuyo contenido no posee dichas características. por ejemplo si un paquete de datos pierde una subcadena de dos caracteres, el paquete no debe tener la subcadena buscada. Esto es porque la eficiencia se basa en asumir que los patrones buscados raramente aparecen en paquetes normales,está búsqueda es susceptible de ataques algorítmicos donde el atacante manipula inteligentemente el texto.

6.2.3.1. Filtrando texto

Un método común de filtrado de texto es el método Bloom [77]. Un filtro Bloom vacío en un arreglo de m bits todo inicializados a 0 ver fig. 6.2. Debe haber definidas k funciones de Hash diferentes, cada una de las cuales aplica un valor clave a cada una de las m posiciones del arreglo. Para agregar un elemento, se alimenta este a cada una de las k funciones de Hash para conseguir k posiciones del arreglo. Se inicializan los bits de esas posiciones a 1.
Para interrogar por un elemento (comprobar si está en 1), se alimenta a cada k función Hash para conseguir k posiciones del arreglo. Si alguno de los bits de esas posiciones son 0, significa que el elemento no está en el conjunto, si este estuviera entonces todos los bit deberían estar en 1, cuando este fue insertado. Si todos son 1, entonces el elemento podría estar en el conjunto, o los bits han sido puestos a 1 durante la inserción de los otros elementos.

El requerimiento de diseñar k funciones de Hash independientes puede ser prohibitivo para un k muy grande. Para buenas funciones de Hash con una salida amplia, debería haber muy baja correlación entre los campos de salida, una función hash de este tipo puede ser usada para generar múltiples funciones de hash distintas particionando la salida en múltiples campos de

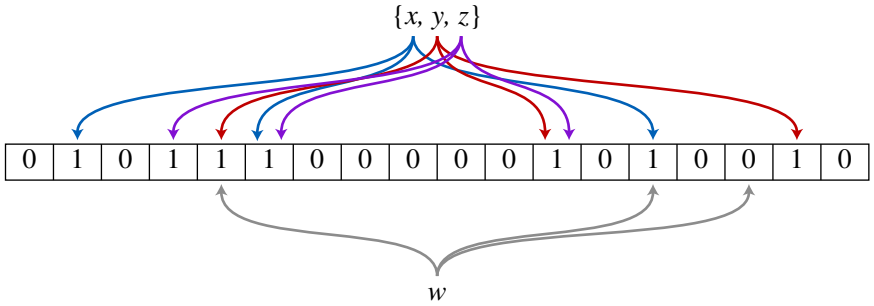

Figura 6.2: Ejemplo de filtrado de texto por método de Bloom

bits.

Alternativamente se puede pasar k diferentes valores iniciales tales como $0, 1, 2, ..., k-1$ a funciones hash que toman un valor inicial, o suman un valor a la clave. Para m o k grandes, la independencia entre funciones hash puede ser relajada con pequeños o despreciables incrementos de falsos positivos o falsos negativos.

Desafortunadamente, quitar un elemento del filtro de Bloom es imposible. El elemento se corresponde a k bits, y aunque estableciendo todos los bits a 0 seria suficiente para quitarlo, esto tiene efectos colaterales de quitar cualquier otro elemento que se corresponda en esos bits, y no hay forma de determinar cuando tal elemento ha sido agregado. Tal quita podría introducir la posibilidad de falsos negativos, los cuales no son permitidos.

La eliminación de un elemento de un filtro de Bloom puede ser simulada usando un segundo filtro de Bloom que contenga items que han sido eliminados. Sin embargo, falsos positivos del segundo filtro se convertirían en falsos negativos del filtro compuesto, los cuales no son permitidos. Esta aproximación limita la semántica de la eliminación ya que el agregado de elementos recién eliminados no está permitido.

Sin embargo está a menudo el caso que todas las claves estén disponibles pero sea muy costoso para enumerarlas (Por ejemplo que requieran lecturas de disco). Cuando la relación de falsos positivos es muy alta, el filtro puede ser regenerado, esto debería ser un evento relativamente poco frecuente. Requerimientos paralelos al filtro de Bloom pueden implementarse en hardware para una mayor eficiencia.

6.3. Tendencias actuales de DPI

Soluciones eficientes para búsqueda de expresiones regulares en DPI está atrayendo considerable interés según describe Bispo [78] el cual presenta varios diseños en hardware, para la búsqueda de expresiones regulares. Algunos de estos diseños pueden alcanzar un rendimiento de varios Gbps. Diseños en software como Snort reclaman haber alcanzado la búsquedas de patrones a velocidades de ciento de miles de firmas por segundo y antivirus como el ClamAV que reclama haber alcanzado 180.000 patrones.Sin embargo de acuerdo a Paxson [79] la búsqueda de expresiones regulares en paquetes es insuficiente debido a la carencia de estado que es indispensable en DPI y espera que gracias al paralelismo de los sistemas actuales se pueda sobrellevar está deficiencia.

Capítulo 7

IDS Distribuidos

7.1. Introducción

En la primera fase del proyecto, se detectaron ataques de inundación de paquetes UDP.
En la segunda fase se detectaron ataques TCP-SYN, para esto se modificaron los módulos de software para adquirir los parámetros del protocolo TCP y los del protocolo ICMP, para detectar inundaciones ICMP.
En base a estas etapas se mejoró el software y se adaptó la red neuronal para optimizar el rendimiento del sistema bajo diversas condiciones de ataque y distintos tráficos de fondo.
En esta etapa se pretende realizar una adquisición de tráfico en múltiples puntos, realizando un preprocesamiento para luego concentrar el análisis en un solo punto. Para llevar a cabo esta tarea se deberán realizar comunicaciones entre procesos, punto de adquisición y puntos de procesamiento, se realizarán modelados de las distintas arquitecturas posibles y simulación del sistema.
Los sistemas NIDS están relacionados con el tráfico de información entre servidores y clientes, típicamente referidos como espías de paquetes (packet-sniffers), estos dispositivos interceptan paquetes que viajan por los medios de comunicación y transportan datos encapsulados en diferentes protocolos tales como, “Frame Relay” o enlaces en Modo de Transferencia Asincrónica (ATM Asynchronous Transfer Mode). Algunos dispositivos NIDS comparan el paquete con

una base de datos de firmas de ataques conocidos y huellas digitales de paquetes maliciosos, mientras que otros analizan la actividad de paquetes buscando un comportamiento anómalo que pueda ser malicioso. En cualquiera de ambos casos un dispositivo IDS debe ser visto principalmente como una defensa perimetral de la red.

De acuerdo a los mecanismos de control un sistema IDS puede ser clasificado como centralizado o distribuido. Un sistema centralizado es aquel en que las tareas de monitoreo y control son realizadas desde un lugar fijo y central tal como el descripto en [45]. La ventaja de un mecanismo centralizado es la facilidad para desarrollarlo y mantenerlo, pero significa un cuello de botella cuando el tráfico en la red es grande y además se convierte en un único punto de falla.

En un sistema distribuido el mecanismo de detección y análisis se puede realizar por diversos agentes cooperantes y autónomos a los cuales se les asigna tareas especificas de detección [46]. La principal ventaja de sistemas distribuidos es que los agentes autónomos trabajan en paralelo y pueden reaccionar en forma más rápida al tráfico mutable de la red.

7.2. Agentes distribuidos

Las redes modernas están distribuidas y los puntos de accesos a Internet son múltiples, esto requiere realizar una verificación en cada punto de accesos a Internet para esto se utilizan agentes distribuidos en dichos puntos.

Se han estudiado distintos tipos de agentes cooperantes y autónomos, llegando a la conclusión que los que mejor se adaptan a la detección de intrusiones en redes son los agentes basados en Sistema de colonia de Hormigas (ACS).

7.2.1. Sistemas de colonias de hormigas (ACS)

La naturaleza ha sido una de las principales medios para formular propuestas y descubrir nuevos paradigmas para resolver diversos problemas en el ámbito de la Ingeniería, principalmente en el ámbito de las técnicas de Inteligencia Artificial entre estos podemos citar:

- Algoritmos Genéticos

- Redes Neuronales
- Colonias de Hormigas

Al analizar el método de cómo las hormigas buscan su alimento y logran establecer el camino más corto desde su nido hasta el alimento, vemos que una hormiga, deposita en su trayecto una sustancia química denominada feromona como una señal odorífera para que las demás puedan guiarse por esta señal. Las feromonas son un sistema indirecto de comunicación química entre animales de una misma especie, que transmiten información acerca del estado fisiológico, reproductivo y social, así como la edad, el sexo y el parentesco del animal emisor, las cuales son recibidas en el sistema olfativo del animal receptor, quien interpreta esas señales, jugando un papel importante en la organización y la supervivencia de muchas especies. Al iniciar la búsqueda de alimento, una hormiga aislada se mueve en forma aleatoria, es decir, sin ninguna señal que pueda guiarla, pero las que le siguen deciden con buena probabilidad continuar por el camino con mayor cantidad de feromonas. Considere la Figura 1 en donde se observa como las hormigas establecen el camino más corto. En la figura 7.1 (a) las hormigas llegan a un punto donde tienen que decidir por uno de los caminos que se les presenta, lo que resuelven de manera aleatoria. En consecuencia, la mitad de las hormigas de dirigirán hacia un extremo y la otra mitad hacia el otro extremo, como ilustra la figura 7.1(b). Como las hormigas se mueven aproximadamente a una velocidad constante, las que eligieron el camino más corto alcanzarán el otro extremo más rápido que las que tomaron el camino más largo, quedando depositado mayor cantidad de feromona por unidad de longitud, como ilustra la figura 7.1(c). La mayor densidad de feromonas depositadas en el trayecto más corto hace que éste sea más deseable para las siguientes hormigas y por lo tanto la mayoría elige transitar por él. Considerando por un lado que la evaporación de la sustancia química hace que los caminos menos transitados sean cada vez menos deseables y por otro lado la realimentación positiva en el camino con más feromonas, resulta claro que al cabo de un tiempo casi todas las hormigas transiten por el camino más corto figura 7.1(d).

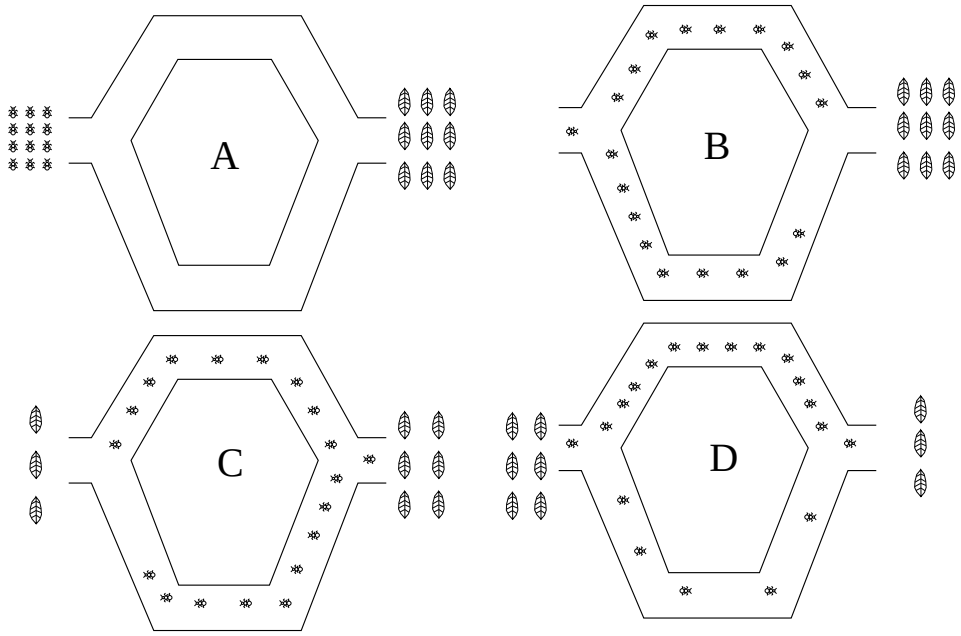

Figura 7.1: Camino de Hormigas.

7.2.2. Algoritmo de agrupamientos de hormigas

El algoritmo del agrupamiento de Hormigas tal como lo describe [80] se a adoptado para el problema de IDS y es el descripto en la Figura 7.2.

El algoritmo (ver figura 7.2) consiste en una población de hormigas que se desenvuelven como agentes autónomos e iterativamente organizan los objetos de datos para un agrupamiento óptimo. Cada objeto de datos se representa como un patrón multidimensional distribuido al azar sobre un espacio de dos dimensiones. En cada escalón de tiempo, cada hormiga busca en el espacio bidimensional moviéndose en forma aleatoria, o saltando con el uso de su memoria a corto plazo. Cada hormiga toma o deja objetos en forma probabilística de acuerdo a la medida de densidad de similitud dada por:

$$f(\Theta_i) = max\left\{0, \frac{1}{S^2}\sum_{\Theta_j \in Neigh_{sxs}(r)}\left[1 - \frac{d(\Theta_i,\Theta_j)}{\alpha(1+\frac{v-1}{v_{max}})}\right]\right\} \tag{7.1}$$

Donde $Neigh_{sxs}(r)$ denota el área local de percepción (S^2 sitios) alrededor del sitio r que una hormiga ocupa en un espacio de dos dimensiones, y la función $d(\Theta_i,\Theta_j)$ mide la distancia

```
begin
//---fase de inicialización---
load_initial_parameters() // v velocidad, valor alfa, radio de
                          // percepción s tamaño de la población
for every data object t(i) do
   randon_place(t(i))    // cada objeto es colocado al azar en el espacio
end for                  //uno por celda
for every ant a(j) do    // cada agente es colocado al azar en el espacio
   randon_place(a(j))    //uno por celda
end for
//--- lazo principal---
for iteration tt = 1 to t(max) do
   for every ant a(j) do
//--- regla 1 ---
     if ((is_carryingObject(a(j)) and is_cellEmpty(r))) then
         compute F(t(i)) and P(drop)
         draw randon number g    // en el intervalo [0,1]
         if (P(drop) > g) then
             drop(a(j),t(i))
             set_unloading(a(j))
             memorize_item(t(i),r,m)   // guarda detalles de t(i) y
         end if                        // localización r en memoria m
     end if
// --- regla 2 ---
     else if ((is_unloanding(a(j))) and (has_object(r))) then
         compute F(t(i)) and P(pick)
         draw randon number g   // en el intervalo [0,1]
         if (P(pick) > g) then
             pick_up(a(j),t(i))
             set_carryingObject(a(j))
             search_memory(t(i),m)  // compara propiedades de los items
                                    // memorizados en m con t(i) y salta
                                    //a la direccion del item más similar
          end if
     end if
//--- regla 3 ---
     wander (a(j),v, N(dir))        // mueve a un nuevo sitio r que  no
   end for                          //esté ocupado por otra hormiga
end for
end
```

Figura 7.2: Algoritmo Camino de Hormigas.

Euclidiana entre dos objetos en el espacio. El mínimo nivel de disparo α corrige las variaciones entre cada par de objetos y la velocidad de movimiento v controla el tamaño del escalón de búsqueda de la hormiga en una unidad de tiempo. La probabilidad de que una hormiga sin carga tome un objeto de dato esta medida por:

$$P_{pick}(\Theta_i) = \left(\frac{k_1}{k_1 + f(\Theta_i)} \right)^2 \tag{7.2}$$

y la probabilidad de que una hormiga llevando objetos lo deje es medida por:

$$P_{drop}(\Theta_i) = \begin{cases} 2f(\Theta_i) & \text{si } f(\Theta_i) < k_2 \\ 1 & \text{si } f(\Theta_i) \geq k_2 \end{cases} \tag{7.3}$$

Donde los niveles de disparo k_1 y k_2 ajustan la probabilidad de tomar o dejar un objeto. Las propiedades de densidad local de similitud medidas en la memoria de corto término son discernidas de la siguiente manera:

- Área de Percepción: El número de objetos de datos percibidos por cada hormiga en el área S^2 es un factor para determinar la compacticidad del agrupamiento y sobre todo el tiempo computacional. Si S es grande, contribuye a una rápida formación de agrupamientos de tal modo que puedan formarse inicialmente agrupaciones menos precisas. Si S es pequeño, contribuye a formar el agrupamiento en forma más lenta, de tal manera que grandes números de agrupamientos precisos pueden ser formados al inicio.

- Factor de escala de similitudes: El factor de escala de similitudes α tiene su valor en el rango de $[0,1]$. Si α es grande la similitudes entre objetos aumenta, de tal forma que es fácil para las hormigas dejar objetos pero es difícil recogerlos. Por lo tanto un mayor número de agrupaciones pueden generarse fácilmente, lo que contribuye a la formación de agrupamientos imprecisos. Si α es pequeña, entonces la similitud entre objetos decre-

ce de tal forma que es relativamente fácil para las hormigas recoger objetos pero difícil depositarlos. Entonces un gran número de agrupamientos pueden generarse fácilmente contribuyendo a la formación de agrupamientos más definidos. Por lo tanto un apropiado valor de α es importante y depende del patrón estadístico de datos.

- Velocidad de movimiento: La velocidad de movimiento V de una hormiga puede ser elegida uniformemente en un intervalo $[1, V_{max}]$. Esta afecta la verosimilitud de tomar o dejar un objeto. Si V es grande, entonces un relativo número de agrupamientos imprecisos pueden formarse inicialmente en gran escala. Si V es pequeño, entonces un número relativamente grande de agrupamientos compactos pueden generarse inicialmente en pequeñas escalas. La velocidad de movimiento es un parámetro crítico para controlar la velocidad de convergencia. Un valor adecuadamente grande de V contribuye a una rápida convergencia.

- La memoria a corto plazo: Cada hormiga se acuerda de los últimos m objetos que ha dejado junto con sus lugares, así que tiene memoria de tamaño fijo m que almacena las propiedades y las coordenadas de los objetos. Cada vez que la hormiga recoge un objeto nuevo, comprueba en su memoria si tiene un objeto similar que ha dejado. Si hay uno, salta a ese lugar (tiene la intención de caer cerca de ese lugar) y evita dejar el objeto en un lugar no visitado. Esta característica mostrada en [81] puede reducir el número de los agrupamientos estadísticamente equivalentes formados en diferentes lugares.

El algoritmo básico de agrupación de hormigas y el modelo mejorado recientemente [81] se evaluaron utilizando los datos KDD-Cup99 del repositorio UCI [82]. En las pruebas se encuentran que sufren de dos problemas importantes en grandes agrupaciones y en redes de datos altamente dimensionales:

En primer lugar, se crean muchos agrupamientos homogéneos y es difícil que se fusionen cuan-

do son de gran tamaño y están espacialmente separados en amplios espacio de búsqueda. Lo cual es muy lento e ineficiente para fusionar agrupaciones.

En segundo lugar, la densidad de similitudes mide solamente características de la formación de agrupaciones en regiones densas locales de objetos de datos similares. Por lo tanto, es ineficaz para dividir conjunto de objetos en los que su diferencia no es muy grande. Como las probabilidades de recoger objetos y de soltarlos están acoplados a esta medida, las agrupaciones se forman en grandes grupos de diferentes clases que pueden ser fusionadas, si los objetos de datos en sus límites son similares. Estos resultados muestran la dificultad de la aplicación directa de agrupamientos de hormigas en la detección de intrusiones en redes. En las secciones siguientes, se adoptará el algoritmo básico con algunas modificaciones [81], haciendo mejoras adicionales para resolver los problemas del Modelo de Agrupamiento de Colonias de Hormigas ACCM (Ant Colony Clustering Model). Estas mejoras se discuten a continuación.

7.2.3. Medida de la entropía regional local

En el Modelo de Agrupamiento de Colonias de Hormigas (ACCM), se propone una combinación de la entropía de la información y el promedio de similitudes, como una métrica adicional a los modelos existentes, para identificar la regiones espaciales de agrupamientos gruesos, agrupamientos compactos y desórdenes de borde de agrupamientos incorrectamente fusionados.

La formula de la entropía de la teoría de la información de Shannon [83], ha sido ampliamente aplicada en muchos campos en la literatura para medir la incertidumbre relativa a una evento, o para caracterizar la impureza de una colección arbitraria de objetos. Si un valor discreto de una variable aleatoria X tiene N salidas $\{x_1, x_2, ... x_n\}$ que ocurren con probabilidades $\{p(x_1), p(x_2), ... p(x_n)\}$, la entropía de la distribución de probabilidad de X está dada por:

$$H(X) = \sum_{i=1}^{N} p(x_i) \log p(x_i) \tag{7.4}$$

El grado de similitud entre cada par de objetos de datos puede revelar sus probabilidades de

agrupación en el mismo agrupamiento. Siguiendo los principios de la entropía de la información de auto-organización y de Shannon, cada hormiga puede medir la impureza de los objetos percibidos dentro de una región local L (de s^2 de sitios) y determinar si el objeto Θ_i en el sitio central de la región L, es igualmente probable que se agrupe con otros objetos Θ_j, utilizando la entropía regional local $H(L)$:

$$H(L) = [g(\Theta_i).\log_2 g(\Theta_i) + (1 - g(\Theta_i)).\log_2(1 - g(\Theta_i))] \tag{7.5}$$

Donde $g(\Theta_i)$ es el promedio de la similitud de objetos en la región L, expresado de la siguiente manera:

$$g(\Theta_i) = \frac{1}{n} \sum_{\Theta_j \in Neigh_{sxs}(r)} \left[0{,}5 + \frac{C(\Theta_i, \Theta_j)}{2}\right] \tag{7.6}$$

En donde n es el número de pares de objetos a ser medidos, $C(\Theta_i, \Theta_j)$ es el coseno de similitudes entre cada par de objetos que puede ser expresado como:

$$C(\Theta_i, \Theta_j) = \frac{\sum_{k=1}^{m} \Theta_{ik}.\Theta_{jk}}{\sqrt{\sum_{k=1}^{m} \Theta_{ik}^2}\sqrt{\sum_{k=1}^{m} \Theta_{jk}^2}} \tag{7.7}$$

Donde Θ_{jk} representa la propiedad K ésima del objeto Θ_i.

Tres ejemplos de configuraciones locales de objetos de datos sobre 9 celdas adyacentes en una grilla son representados en la figura 7.3, en la cual diferentes clases de objetos son mostrados con distintas tonalidades.

Cuando los objetos de datos en la región local están muy próximos al mismo agrupamiento como en la figura 7.3 a, o muy distintos y pertenecen a diferentes agrupaciones como en la figura 7.3 b, la incertidumbre es baja y $H(L)$ es cercana a 0. Interesa una configuración desordenada de objetos como en la figura 7.3 c. Donde $f(\Theta_i)$ o $g(\Theta_i)$ pueden dar un valor medio arbitrario cercano a $0,5$, lo que no puede estimular en forma precisa a las hormigas para que recojan o dejen objetos en el sito central. Sin embargo un valor alto de $H(L)$ puede identificar esta estructura compleja con alto grado de incerteza para agrupar objetos dentro del mismo agrupamiento. Por lo tanto pueden encontrarse agrupaciones compactas que tienen la propie-

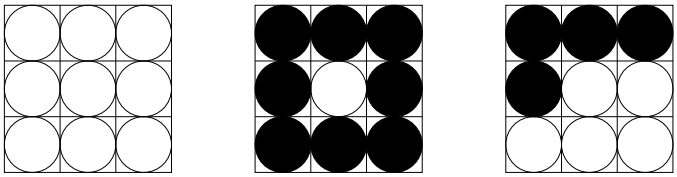

Figura 7.3: Configuración local de objetos.

dad de un alto $g(\Theta_i)$ y bajo $H(L)$, agrupaciones imprecisas que tienen bajo $g(\Theta_i)$ y bajo $H(L)$ y agrupaciones fusionadas incorrectamente con desordenes de bordes, tienen alto $H(L)$ como se ve en figura c. Esas propiedades contribuirán a las siguientes mejoras.

7.2.4. Infraestructura de feromonas

La formación de grupos más grandes y la destrucción de las pequeñas agrupaciones de hormigas de acuerdo al algoritmo de agrupación se basa en la retroalimentación positiva y negativa de la auto-organización. La retroalimentación referenciada en el algoritmo se discute en el trabajo de Theraulaz [84].

Con el fin de construir un modelo sólido de agrupaciones utilizando estos efectos colaborativos, es necesario equilibrar las fuerzas correspondientes de exploración y explotación para la optimización global.

Muchas especies de insectos sociales hacen uso de algunas sustancias químicas llamadas feromonas para comunicarse indirectamente para el logro de algunos comportamientos colectivos, tales como la construcción de nidos y la búsqueda de comida. La optimización global de las conductas inteligentes en estos enjambres es el criterio esencial para sus necesidades de supervivencia.

En el ACCM, la formación de agrupaciones y el objeto de la búsqueda son consideradas para la construcción de nidos y la recolección de alimentos, respectivamente. Las feromonas son introducidos para este tipo de estrategias de búsqueda: grupos de feromonas guían a las hormigas para buscar agrupaciones compactas, mientras que objetos de feromonas guían a las hormigas a buscar los objetos para ser recogidos. Sus interacciones colaborativas se formulan de la

siguiente manera:

- Regla de deposición de Feromonas: Un grupo de hormigas deposita feromonas en una grilla después de que deja caer sus objetos Θ_i con éxito. La intensidad de las agrupaciones de feromonas depositadas en el sitio j por las hormigas en m colonias en el tiempo t, se calcula como:

$$r_{ij}(t) = \sum_{i=1}^{m} \left[C_j.(1 - H(L)).g(\Theta_i) \right] \tag{7.8}$$

  Donde C_j es la cantidad de feromona depositada por la hormiga en el sitio j. Por otra parte, una hormiga deposita feromonas en la grilla j después de que coge un objeto Θ_i con éxito. La intensidad de la objetos de feromonas en el sitio j depositados por m colonias de hormigas en el tiempo t, se calcula como:

$$r_{ij}(t) = \sum_{i=1}^{m} \left[\Theta_j.H(L) \right] \tag{7.9}$$

  Donde el Θ_i es la cantidad de objetos feromona depositados por las hormigas en el sitio j. Una fuerte agrupación de feromonas en un sendero atrae a más hormigas transportando objetos para buscar agrupaciones compactas con objetos similares, mientras que un objeto con feromonas más intenso en el sendero atrae a las hormigas que no portan objetos para buscar agrupación de objetos con alto $H(L)$ y también estimular a disminuir su similitud de escala de valor α (1) de tal manera que puede recoger los objetos de datos fácilmente.

- Probabilidad de transición: La probabilidad de transición de una hormiga para pasar de su sitio actual i a un sitio j próximo, se mide como:

$$P_{ij}^{k} = \begin{cases} \frac{1}{n} & \text{si } \sum_{j=1}^{n} \left[\tau_{ij}(t) \right]^{\gamma} = 0 \, \forall \, j \in N_{dir} \\ \frac{\left[\tau_{ij}(t) \right]^{\gamma} + \beta \sum_{j=1}^{n} \left[\tau_{ij}(t) \right]^{\gamma}}{(\beta+1) \sum_{j=1}^{n} \left[\tau_{ij}(t) \right]^{\gamma}} & \text{De otra forma} \end{cases} \tag{7.10}$$

En donde τ es la matriz de feromonas, $\tau_{ij}(t)$ es la intensidad de la feromona en el sitio j en el tiempo t, el N_{dir} es un conjunto de n posibles escalones y β y γ ambas controlan la dependencia relativa del camino de feromonas. Si una hormiga no detecta la intensidad de las feromonas en sus sitios vecinos, ésta se mueve en forma aleatoria. De lo contrario, probabilísticamente se mueve al sitio con mayor intensidad de feromona. El parámetro de escala estático γ, esta en el rango de $[0,1]$ y determina la sensibilidad a la intensidad de feromona. Si este es grande, el sitio de alta intensidad de la feromona tiene más probabilidades de ser seleccionado. Si es cercana a cero, la probabilidad de elegir cualquiera de los sitios de vecindad tiene una distribución uniforme. Su valor se establece en $0,7$ para las pruebas. El parámetro β, de escala que va desde $[0,1]$, es la relación de caídas exitosas de los objetos transportados por una hormiga cada 2500 iteraciones, el cual influye en sus comportamientos de búsqueda. Si la tasa es alta, demuestra que la hormiga puede buscar bien de acuerdo a su velocidad y memoria a corto plazo, la influencia de la intensidad de las feromonas será relativamente pequeña. Si la tasa es baja y cerca de cero, demuestra que la hormiga necesita una búsqueda guiada, lo más probablemente es que se mueva a un sitio vecino con alta intensidad de feromona.

- Regla Global de Actualización de Feromonas : Cuando se inicializa el modelo, toda intensidad de feromona en la matriz τ se establece en 0. En la realización de una iteración t, las feromonas se desintegran por la evaporación y la difusión. La intensidad de la feromona en el sitio j observada por una hormiga en el sitio i en el tiempo $(t+1)$, es:

$$\tau_{ij}(t+1) = \rho.\tau_{ij}(t) + r_{ij}(t) + q_{ij}(t) \tag{7.11}$$

Donde $\rho \in [0,1]$ es el coeficiente de evaporación. La función de difusión propuesta por [85] es adoptada de la siguiente forma:

$$q_{ij}(t+1) = \sum_{j \in N(j')} \frac{F}{\|N(j')\|} \left[r_{ij}(t) + q_{ij}(t) \right] \tag{7.12}$$

Donde $F \in [0,1]$ es el parámetro de propagación y $N(J')$ es un conjunto de vecinos del sitio j. La infraestructura guía a las hormigas a la mejor ruta con probabilidad proporcional tanto a la intensidad como al tipo de ensayo con feromonas para el agrupamiento óptimo con fusiones y divisiones. La realimentación positiva y negativa es controlada por las propiedades de los grupos de datos durante todo el proceso de agrupación.

7.2.5. Modificación de la memoria a corto plazo y adaptación α

La memoria a corto plazo se modifica en [80]. Se memorizan propiedades adicionales de los objetos, como entropía regional local y la similitud promedio de éxito de los objetos depositados. Después de que una hormiga recoge con éxito un nuevo objeto, aprovechando los datos de su memoria, salta probabilísticamente a un grupo compacto donde la ubicación del objeto almacenado tenga un mayor $g(\tau_i)$ pero más bajo $H(L)$ llevando el objeto allí. Esto reduce estadísticamente agrupaciones equivalentes en grandes espacios de búsqueda. Además, desde el valor de α en (1) se aumenta la escala de densidad de la medida de similitud, por un régimen de adaptación propuestas en [81], el cual es adoptado y ajustado como sigue: El parámetro α de cada hormiga puede ser actualizado mediante la regla:

$$\alpha \leftarrow \begin{cases} \alpha + 0,01 & \text{si } r_{fail} > 0,9 \\ \alpha - 0,01 & \text{si } r_{fail} \leq 0,9 \end{cases} \tag{7.13}$$

Donde r_{fail} es la relación de acciones de dejar objetos fallada en 200 iteraciones. Si r_{fail} es muy grande, el valor de α y la similitud entre objetos, disminuirán de tal manera que una hormiga pueda dejar fácilmente objetos.

7.2.6. Esquema de selección, configuración de parámetros y adquisición de agrupaciones

Equilibrar la presión selectiva y la diversidad de la población en el mecanismo de muestreo, es un factor importante en el diseño de algoritmos evolutivos. Un sistema de selección de

corridas se propone contrarrestar la diversidad de la población y encontrar los valores óptimos de los parámetros de control como el valor α, la velocidad y radio de percepción en el tiempo. El rendimiento del comportamiento de la hormiga puede identificar si ésta es elitista en la colonia. En cada iteración t, el rendimiento p de cada hormiga se puede medir como:

$$p_t = \begin{cases} [g(\Theta_i) + 1 - H(\Theta_i)]/2 & \text{si la acción dejar es activada por } P_{drop}\, y\, f(\theta_i) \\ [1 - g(\Theta_i) + H(\Theta_i)]/2 & \text{si la acción tomar es activada por } P_{pick}\, y\, f(\theta_i) \\ 0 & \text{si la acción dejar o tomar no es activada} \end{cases} \tag{7.14}$$

El rendimiento promedio P de una hormiga cada 5000 iteraciones es calculado como:

$$P = \sum_{i=1}^{5000} p_t/n \tag{7.15}$$

El cual es la medida de construir agrupaciones compactas y destruir agrupaciones no compactas en diferentes estadios del proceso de agrupaciones dinámicas, y n es el número total de de tomas y abandono de objetos con éxito. Usando 5000 iteraciones como generación, todas las hormigas se comparan en términos de su rendimiento promedio. Los valores de los parámetros de las hormigas elitistas con un mayor rendimiento permanecen sin cambios en la próxima generación, mientras que los valores de las hormigas de bajo rendimiento heredan los parámetros de las hormigas elitista en cada generación. Para evitar la fuerte presión selectiva que produce la convergencia prematura de la búsqueda y la pérdida de la diversidad, se establece un tamaño pequeño de corridas igual a dos. Esta estrategia simplifica la configuración de los parámetros y evita la rigidez del modelo que se está manejando por sus valores constantes. Al comienzo de la ejecución, a cada hormiga se le da diferentes valores de parámetros iniciales, que son elegidos de manera uniforme en intervalos acotados. Para la muestra de datos KDD Cup99 de IDS, la configuración inicial de los parámetros son: tamaño de la cuadrícula: $460 * 460$, tamaño de la población: 288, velocidad: $v \in [1, 150]$, radio de la percepción: $s \in (3, 5, 7, 9)$, factor de escala de similitud:$\alpha \in (0, 1)$, tamaño de la memoria a corto plazo: $m = 20$, número máxi-

mo de iteraciones en cada ejecución: $t_{max} = 106$, y las constantes de umbral para P_{pick} y p_drop: $K_1 = 0,1, k_2 = 0,3$. Se adopta un algoritmo jerárquico de agrupación ponderado de enlace único como en [81], para recuperar las agrupaciones concretas de los grupos separados espacialmente en el ACCM.

7.2.7. Resultados experimentales

En las pruebas se estudian las características del ACCM mediante la evaluación de su validez de agrupación y rendimiento de clasificación. El ACCM se compara con los algoritmos existentes basados en la agrupación de hormigas, K-Means y E-M, para el análisis de rendimiento. En particular el desempeño del ACCM, que se propone para resolver los complicados problemas de detección de intrusos en redes, es intensamente evaluado por medio del conjunto de datos de referencia. Por otra parte, se examinan la eficacia de la aplicación de diferentes métodos de extracción en función de los pasos de preprocesamiento del ACCM para la detección de intrusiones.

En este caso no se pudieron utilizar la base de datos obtenida del Firewall de la Universidad Nacional de Córdoba como en los capítulos anteriores, donde se superponía al trafico normal, ataques de inundación generados para ese propósito, el tráfico normal puede contener ataques, al no poseer un IDS patrón para evaluar este tráfico no se puede utilizar para una detección mas fina de intrusiones, por esta causa se busco un conjunto de datos de prueba normalizados.

7.2.8. Conjunto de datos, descripción y preprocesamiento

Seis conjuntos de datos de pruebas del mundo real están disponibles en el repositorio UCI [82] y son tomados para evaluar la referencia de base, para el sistema del ACCM. De las seis bases de datos se usa la KDD-Cup99 de datos IDS.

La base de datos KDD-Cup99 de detección de intrusos, es un conjunto de datos que se utiliza ampliamente como datos de referencia para evaluar los sistemas de detección de intrusiones, ésta es relativamente compleja, de gran escala, de alta dimensión y alto ruido. Además da

un alto grado de desequilibrio de clase y éstas clases son muy superpuestas. En las pruebas, se aplica un 10 % de datos que contiene 494021 registros de entrenamiento. Cada registro de conexión representa una secuencia de la transmisión de paquetes de inicio y fin en un período de tiempo, y se pueden clasificar como normal, o una de las 22 clases diferentes de ataques. Todos los ataques se dividen en 4 categorías principales.

1. Denegación de servicio (DoS) - La denegación de los servicios que se acceden por los usuarios legítimos, por ejemplo, inundaciones SYN (Neptuno) y los ataques LAN.

2. Remote to Local (R2L) - El acceso no autorizado desde una máquina remota, por ejemplo, adivinar la contraseña y ataques de escritura ftp.

3. Usuario a root (raíz U2R) - El acceso no autorizado a ganar los privilegios Super usuario (root), por ejemplo, buffer overflow y los ataques de desbordamiento de perl.

4. Sonda (Probe) - Vigilancia y exploración de recolección de información, por ejemplo, escaneo de puertos y ataques nmap.

Para evitar el deterioro del rendimiento debido al problema del desequilibrio en la formación de clases, se aplica al azar un sub-método de muestreo a las tres grandes clases: “Normal”, “Smurf” y “Neptuno”, que incluyen el 98 % de los registros del conjunto de datos de entrenamiento. Los datos de entrenamiento nuevos, contienen 10^4 registros de la clase normal y 10^3 registros para cada una de las clase Neptuno y Smurf, mientras que el número de registros de otras clases, permanece intacta. Como resultado, un total de 20.752 registros son aplicados para el entrenamiento del ACCM. Para hacer la tarea de detección más realista, el modelo de entrenamiento, es evaluado utilizando datos de prueba KDD-Cup99 independientemente, que tiene 311029 registros con diferentes clases de distribución de probabilidad y otros 14 tipos de ataque no identificados. Las distribuciones de datos de entrenamiento detalladas y los conjuntos de datos de prueba se muestra en el cuadro 7.1. Como los registros de conexión de red contienen características tanto continuas como nominales, las características nominales como el protocolo (TCP / UDP / ICMP), tipo de servicio (http / ftp / telnet /...) y el indicador del estado de TCP (SF / REJ /...) son primero convertidos a características numéricas binarias, por lo

Cuadro 7.1: Distribución de los datos en KDD-Cup99.

			Original		Datos de entrenamiento	
Índice de Clase	Nombre Clase	Categoría Ataque	Num. de Instancia	Distr. Aprox.	Num. Instancia	Dist. Aprox.
0	Normal		97278	16.69 %	10000	48.19 %
1	Back	DOS	2203	0.44 %	2203	10.62 %
2	Buffer Overflow	U2R	30	0.006 %	30	0.145 %
3	escritura ftp	R2L	8	0.002 %	8	0.039 %
4	búsqueda contraseña	R2L	53	0.011 %	53	0.255 %
5	imap	R2L	12	0.002 %	12	0.058 %
6	ipsweep	PROBE	1247	0.252 %	1247	6.009 %
7	land	DOS	21	0.004 %	21	0.101 %
8	loadmodule	U2R	9	0.002 %	9	0.043 %
9	multihop	R2L	7	0.001 %	7	0.034 %
10	neptune	DOS	107201	21.7 %	1000	4.819 %
11	smurf	DOS	280790	56.84 %	1000	4.819 %

tanto, un total de 123 funciones numéricas se construyen para el cálculo numérico, tales como la extracción de características.

7.2.9. Métodos para extracción de características del ACCM para IDS

Para mejorar el agrupamiento y resolver el problema de la multidimensionalidad de las características de los datos capturados de las redes, se han evaluado cuatro algoritmos no supervisados de extracción de características:

- Regla de análisis de componentes PCA (Principle Component Analisys) [86] que aplica funciones estadísticas de segundo orden para extraer Reglas de Componentes PC (Principle Componen) como combinaciones lineales ortogonales de las características de origen, para reducir la dimensionalidad. En el trabajo se aplica PCA para remover los valores extremos y ajustar el proceso de agrupamiento para la reducción de dimensiones. Los algoritmos K-means y E-M, como así también el del ACCM es entrenado con diferentes

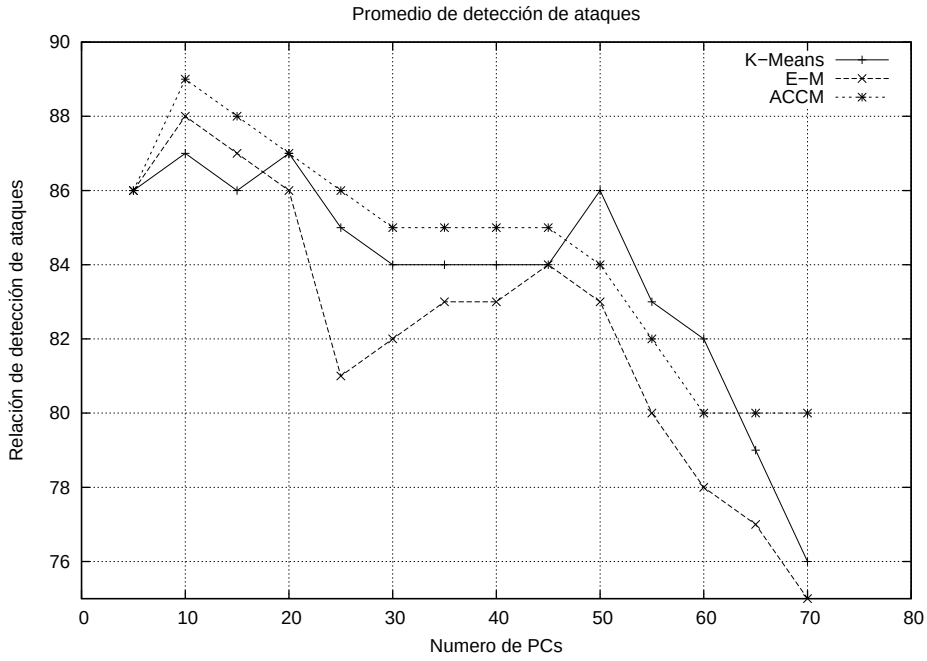

Figura 7.4: Promedio de detección de ataques.

PCs extraídos de los datos de entrenamiento y evaluados con los datos de prueba. Los resultados obtenidos en términos del promedio de la relación detección de ataques ADR (Attack Detection Rate) son mostrados en la figura 7.4.

En la figura 7.5 se muestra el promedio de relación de falsos positivos FPR (False Positive Rate). Se encontró que usando 8 PCs con valores propios suficientemente grandes se obtiene un buen rendimiento en ambos valores. Del gráfico se desprende que los valores de E-M se vuelven inestables cuando los valores extremos se incrementan con el número de dimensiones. Por el contrario los del ACCM se comportan de manera más estable que los de los algoritmos K-Means y E-M cuando aumenta el número de PCs, implicando que el ACCM es insensible al ruido de los datos, el cual existe en el tráfico de redes.

- Análisis de Componentes Independientes ICA (Independent Component Analysis): [87] La independencia estadística que puede ser medida con estadísticas de orden superior, es generalmente una propiedad más fuerte que la ofrecida por la PCA, para extraer características de datos sin rotular. El ICA ha recibido últimamente mucho interés en áreas de la Ingeniería Biomédica en procesamiento de imágenes y procesamiento de señales.

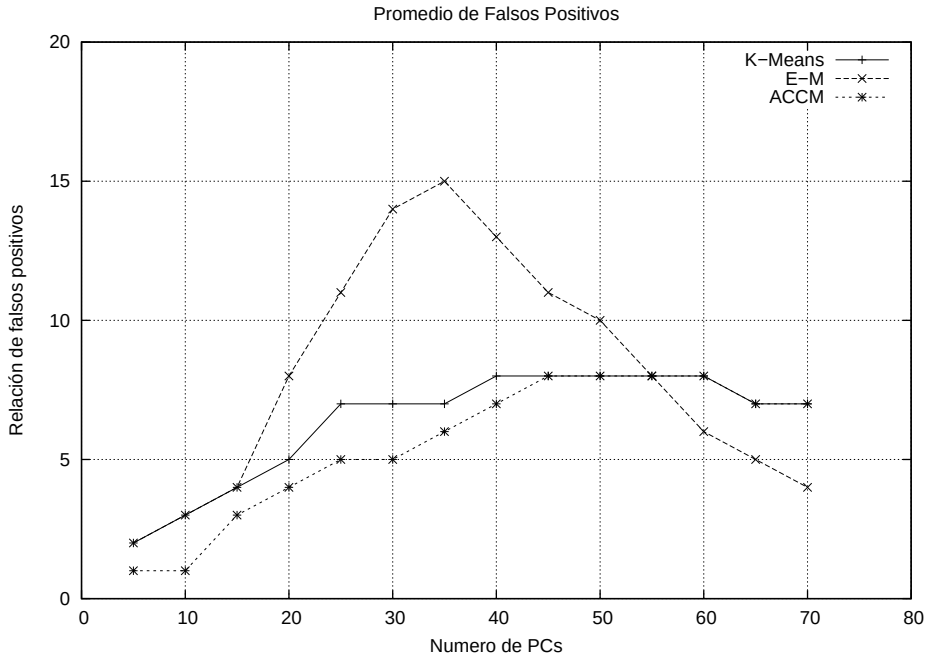

Figura 7.5: Promedio de falsos positivos.

ICA descompone los datos en sus componentes fuentes que son estadísticamente independientes unas de las otras, tanto como sea posible. ICA ha probado ser efectiva para extraer características de distribuciones no gausianas. Estudios recientes han mostrado que el tráfico de redes tiene un distribución no Gausiana y una distribución de Poisson. En este caso aplicando ICA se ha logrado extraer característica de las redes de datos latentes. Tres algoritmos bien conocidos han sido evaluados en las pruebas: InfomaxICA [88], Infomax Extendido [89] y FastICA [90].

El PCA ha sido aplicado como paso de preprocesamiento para los diferentes algoritmos ICA, para reducir el ruido de los datos y evitar un sobre entrenamiento en ICA.

Determinar el número de componentes independientes IC (Independ Componentes) a ser estimadas en ICA es una tarea difícil y requiere de la intervención humana, los 8 PCs encontrados previamente son usados de entrada para el ICA.

Después de que ICA y PCA, se aplican como preprocesos, el ACCM aplica la función coseno de similitudes (7) en lugar de la distancia Euclidiana, para determinar que método de extracción de características mejora el rendimiento en la agrupación de datos en redes, en el cuadro 7.2 se

Cuadro 7.2: ADR y FPR para los diferentes algoritmos.

	K-Means		E-M		ACCM	
Usando 8 PCs/ICs	ADR	FPR	ADR	FPR	ADR	FPR
PCA	87.27	3.32	88.14	4.07	88.39	1.35
InfomaxICA	88.95	4.64	89.08	5.81	91.68	2.79
Ext-InfomaxICA	88.51	3.76	89.12	4.16	92.07	1.17
FastICA	89.17	4.29	90.94	4.24	92.23	1.53

Cuadro 7.3: Matriz de Confusión de los datos de entrenamiento.

Clase y N. de patrones	DOS	U2R	R2L	Prueba	Normal	Reuso (%)
DOS (5467)	5393	0	10	24	40	98.655
U2R (52)	0	28	17	0	7	53.85
R2L (1126)	25	13	958	29	101	85.08
Prueba (4107)	57	0	8	3952	90	96.23
Normal (10000)	74	0	52	43	9831	98.31
Precisión (%)	97.19	68.29	91.67	97.63	97.64	
Medida F	97.91	60.22	88.25	96.92	97.97	

muestra los diferentes algoritmos empleados.

Los resultados del cuadro 7.2 muestran que FastICA mejora el ADR, pero degrada el FPR, comparado con PCA y Ext-InfomaxICA. En términos de ADR, no hay una diferencia significativa entre Ext-InfomaxICA y FastICA. Como el impacto de una alta FPR es más crítico cuando el tráfico de red es mucho mayor que el intento de intrusiones en las redes, Ext-InfomaxICA es elegido como el método de extracciones de características porque obtiene el menor FPR.

La Matriz de confusión del cuadro 7.3, muestra el que el ACCM es capaz de agrupar los ataques de intrusión en diferentes grupos, particularmente para las clases DOS, R2L y ataques de prueba. Adicionalmente, una gran cantidad de conexiones normales son agrupadas en grupos normales.

7.2.10. Arquitectura Multi Agente IDS

Las redes modernas tienen hoy en día múltiples puntos de acceso. Desplegar un sistema IDS centralizado no logra proteger la red en toda su extensión, además una red centralizada cuando trabaja bajo grandes cargas de tráfico, no puede detectar intrusiones eficientemente, principalmente en redes conmutadas.

Un sistema Multi Agente MAS (Multi-Agen System) puede ofrecer un sistema descentralizado eficiente, y un mecanismo de control para una detección de intrusiones en redes en gran escala. A cada agente autónomo MAS se le puede asignar diferentes tareas IDS, ejecutándose en diferentes subredes. Una arquitectura Multi Agente basada en el ACCM es propuesta a continuación. La estructura jerárquica del sistema es mostrado en la Figura 7.6, en el cual cada nodo persigue un objetivo orientado a agentes.

Las responsabilidades de cada agente se definen a continuación:

- Agentes Monitores. Son los encargados de capturar los paquetes de la red de datos, reduciendo el ruido irrelevante y extrayendo las características relevantes latentes. Como los agentes están ubicados en subredes solo tienen información parcial de los posibles ataques y no pueden detectar una intrusión generalizada. Por lo tanto ellos envían información preprocesada a los agentes de decisión para un análisis más integral. Estos agentes capturan datos en modo promiscuo.

- Agentes de decisión. Estos agentes llevan a cabo la tarea de aprendizaje de anomalías y preclasificación para los datos suministrados por los agentes monitores.

- Agentes de Acción. Realizan una acción pasiva o reactiva para responder a los diferentes ataques notificados por los agentes de decisión.

- Agentes de Coordinación. Estos agentes aportan análisis a alto nivel desde los agrupamientos de agentes de decisión, en orden a mejorar la eficiencia y la detección distribuida.

- Agentes de Interfase de usuario. Son los encargado de interactuar con los usuarios y operadores de redes.

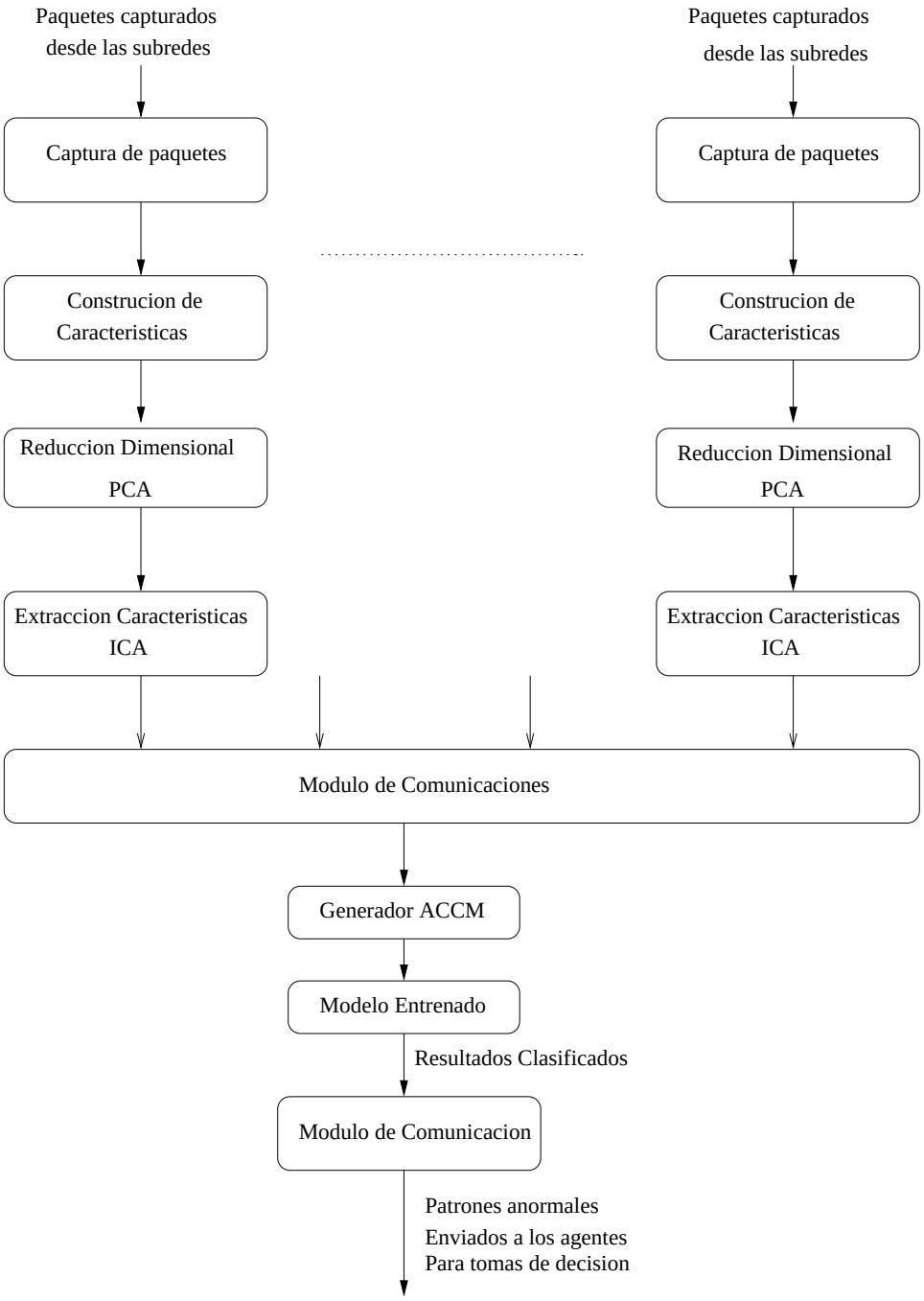

Figura 7.6: Estructura interna del detector de intrusiones.

- Agentes de Registro Estos ubican todos los agentes autónomos para facilitar la dinámica del MAS. De esta forma cada agente puede ser creado o removido de una sub red. Actúa como intermediario y organiza los canales de comunicación entre agentes.

7.3. Conclusiones

Los ataques crecen en complejidad y sofisticación, adoptando formas polimórficas, esto demanda cada vez más IDS con mejores rendimientos y más inteligentes. A partir de la colonia de hormigas se ha presentado un sistema de auto aprendizaje no supervisado el ACCM para la detección de intrusiones en redes. El ACCM mejora los algoritmos de colonias de hormigas existentes buscando una heurística de agrupamiento cercana a la óptima. Se han introducido conceptos tales como entropía regional local,infraestructura de feromonas, memoria modificada a corto termino, selección de corridas, balance entre exploración y explotación, optimización entre unión y separación de agrupamientos. El ACCM determina automáticamente el número de agrupaciones crítica requerida como entrada para los algoritmos K-Means y E-M. El ACCM es insensible para los valores extremos, los cuales existen en el tráfico de redes y corrompen la estructura de agrupamientos.

Los resultados de las pruebas demuestran que en comparación con los algoritmos de hormigas existentes, el ACCM reduce significativamente el número de agrupaciones estadísticamente equivalentes y genera una solución de agrupamientos cercana a la óptima. En general la calidad de agrupaciones y el rendimiento de clasificación del ACCM es respectivamente más estable y preciso que K-Means y E-M, como se mostró en las tablas. Los resultados obtenidos demuestran que la aplicación del ACCM junto con el algoritmo Ext-InfomaxICA, es efectiva para detectar ataques conocidos y nuevos en redes, con una relación de detección alta y con reconocimiento de tráfico normal con una baja tasa de falsos positivos. Finalmente se propone una arquitectura de multi agentes distribuidos basados en el ACCM para detectar intrusiones en grandes redes.

Capítulo 8

Herramientas de Prueba de IDS

8.1. Introducción

Las herramientas de prueba de IDS son indispensables para evaluar el comportamiento de estos simulando el ambiente en que van a trabajar.

Las principales herramientas que se encuentran en la mayoría de los laboratorios son los generadores de tráfico (TG Traffic Generator) y las herramientas de pruebas de evasión.

Los generadores de tráfico pueden generar tráfico Normal o tráfico malicioso, de tal forma de simular los diferentes escenarios en que trabajará un IDS.

Existe una gran variedad de TG en este capítulo presentaremos algunas de libre disponibilidad en internet y otras propietarias, dentro de estas existen TG implementados en hardware, y otros con un programa que puede correr en diversas plataformas. Los TG son diseñados para generar paquetes de tráfico en un solo sentido desde una fuente a un sumidero en un segmento de red. Los protocolos usados son TCP, UDP, ICMP y ARP en orden a probar diversos escenarios.

El principal propósito de TG es para ser usado en un laboratorio en un ambiente de prueba controlado, para comprobar NIDS, tanto aquellos basados en software como Snort y Bro como aquellos basados en hardware. Bajo estas condiciones es posible evaluar que sistemas son más efectivos para detectar tráfico malicioso como la mezcla de tráfico normal y tráfico malicioso afecta la capacidad de detección, y con que anchos de banda puede operar eficientemente.

8.2. Modo de operación de los TG

La construcción de los TG se basan en el uso de reglas de dominio público, tales como las usadas en los NIDS Snort, que comprende una base de datos de patrones de tráfico que son sospechosos de ser maliciosos, estas bases de datos son continuamente puestas al día con los patrones de nuevos ataques.
La información contenida en estas base de datos, es organizada en forma de reglas, donde cada una es expresada en un formato fijo que provee toda la información necesaria para describir el tráfico malicioso.
Los pasos para generar el tráfico malicioso son los siguiente:

- Un programa analiza sintácticamente la reglas contenidas en la base de datos y almacena las en las variables de la estructura de datos que será usada para alimentar el canal de datos, ver fig. 8.1

- Otro programa es ejecutado desde la linea de comandos con parámetros tales como; modo de transmisión, protocolo, número de paquetes a enviar e Ip del destinatario.

Cuando es preciso generar patrones específicos de tráfico a altas velocidades se realizan programas específicos generalmente en C. las ráfagas de tráfico de ataque se pueden mezclar mediante un conmutador de red, con tráfico normal.

8.3. Antecedentes de TG

Existe una gran variedad de productos comerciales como productos GNU en los siguientes párrafos se describen algunos de los equipos y software más usados en los laboratorios de pruebas de NIDS.

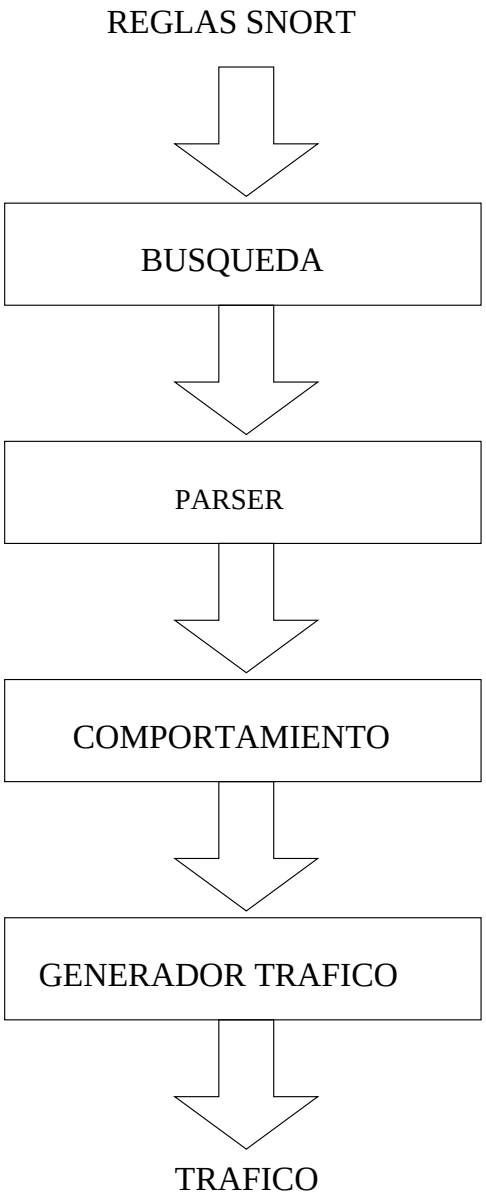

Figura 8.1: Proceso de generación de tráfico.

8.3.1. TG Comerciales

Los TG comerciales generalmente son equipos implementados en hardware y debido a que no son de producción masiva son de alto costo los principales fabricantes son:

8.3.2. BreakingPoint

Las herramientas de pruebas de BreakingPoint [91] proporcionan capacidades de pruebas resistencia a la recuperación mediante la simulación realista de tráfico de aplicaciones combinadas se mezcla con los ataques de seguridad en vivo, velocidad de flujo en linea y condiciones de carga máxima. Las herramientas de pruebas de capacidad de recuperación proporcionan:

- Prueba todas las vulnerabilidades conocidas, incluyendo ataques a la seguridad actual y global y protocolos de aplicación difusos para endurecer las defensas.
- Generación de tráfico a velocidades de línea y sesiones de aplicaciones mixtas con aplicaciones reales para probar las vulnerabilidades y problemas de rendimiento bajo condiciones de carga máxima.
- Herramientas automatizadas que son constantemente actualizado con las vulnerabilidades de seguridad más recientes y los protocolos de aplicación.
- Soporte para crear y probar los protocolos de aplicación de propietarios y ataques de seguridad personalizada.
- Capacidad para garantizar el cumplimiento con los últimos estándares de reglamentación y la industria.
- Fácil de implementar pruebas automatizadas y buena interfaz de usuario, ayudando a reducir el tiempo de prueba en un 50 % o más.

Entre los principales productos de BreakingPoint se destaca el BreakingPoint Elite el cual proporciona los componentes necesarios para probar la resistencia, el realismo , el rendimiento del período de sesiones, el rendimiento y la facilidad de uso, incluyendo:

- 15 millones de sesiones concurrentes TCP y 1,5 millones de sesiones TCP por segundo
- 40 Gigabits por segundo de Nivel 4-7 de tráfico en un chasis de 3-slot
- 80 Gigabits por segundo de tráfico de Nivel 2-3 en un chasis de 3-slot
- Se pueden combinar fácilmente con múltiples BreakingPoint Elite de tal forma de proporcionar una cantidad ilimitada de tráfico de Nivel 2-7 , manteniendo una única interfaz de configuración de prueba y la presentación de informes
- Capacidad para mezclar interfaces de 1Gigabit Ethernet y 10Gigabit Ethernet , y escala a grandes entornos de prueba
- Una interfaz de gestión intuitiva, capacidades multi-usuario, integrada de capas 2-7 presentación de informes y completa automatización de las pruebas
- Más de 75 protocolos de aplicación que pueden ser mezclados con más de 4.200 ataques a la seguridad, una API para integrar el tráfico de aplicaciones propietarias.

8.3.3. Candela Technologies

Candela Technologies [92] produce una gran variedad de TG desde modelos de baja capacidad que generan tráfico a una velocidad de 45 Mbps sobre interfases DS3 y T3 como el modelo LANforge ICE, hasta equipos con capacidad de generar tráfico a una velocidad de 5 Gbps con memoria de estado como el modelo LANforge FIRE & ICE: 5.1.1 el cual soporta envío de replicas de paquetes, generación de tráficos de VOIP.

8.3.4. TG GNU

TG Es un proyecto desarrollado en colaboración con SRI International [93] con mejoras financiadas por USC/ISI Postel Center para la experimentación en redes. TG es un generador de tráfico que crea un flujo de datos TCP y UDP unidireccional entre una fuente y un sumidero. El tráfico es descripto en términos de los intervalos de tiempo y la longitud de los paquetes. La

información relacionada con el origen y destino de los paquetes, el tiempo de transmisión y el tiempo de recepción son almacenados en archivos binarios para su posterior procesamiento por por el programa “dcat”. Dcat procesa los archivos binarios y produce una representación leíble en formato “ASCII”. Un programa en Perl transforma estos datos en forma apropiada para poder ser visualizada con programas de dominio publico tales como xplot, xgraph y gnuplot.

8.3.5. Swing

Swing es un generador de tráfico [94] a lazo cerrado, captura las interacciones de paquetes de un rango de aplicaciones usando un simple modelo estructural. Comienza observando el tráfico en un único punto de la red, automáticamente extrae la distribución por usuario, aplicación y comportamiento de la red y genera tráfico verdadero correspondiente a los modelos y ambientes de red que se están emulando, ejecutando los protocolos de red correspondientes.
Para lograr estos los desarrolladores se enfrentaron a dos desafíos, primero se requiere un modelo base con una semántica simple con parámetros significativos, que especifiquen en forma completa una traza dada. Por semántica significativa se refiere a la posibilidad de relacionar aplicaciones de alto nivel a modelos de condiciones de la red. Segundo se requiere una técnica para construir el modelo desde las trazas de paquetes existente, para validar la eficacia de la captura de paquetes.

8.3.6. Earlybird

Earlybird es un generador de tráfico con una huella de gusano [95]. Se ha focalizado en el análisis y modelizado del comportamiento de Internet como resultado del desarrollo de otras herramientas tales como HARPOON generador de trafico a nivel de flujo, SURGE generador de tráfico web y MACE generador de tráfico malicioso y análisis de camino crítico para transacciones TCP. Desarrollado por el Departamento de Ciencias de la Computación de la Universidad de Wisconsin Madison, poniendo el foco en la comprensión de los detalles de transporte y

comportamientos de rutas, la naturaleza de la perdida de paquetes y las aplicaciones de análisis de las técnicas de multi resolución para medir paquetes de datos.

8.3.7. Harpoon

Harpoon es un generador de tráfico a nivel de flujo [96]. Este usa un conjunto de parámetros de distribución que pueden ser automáticamente extraídos de Netflow que presentan las mismas característica estadísticas que las medidas en Internet, incluidas las características espaciales y temporales. Harpoon puede ser utilizado para generar tráfico de base para aplicaciones o pruebas de protocolos, para probar conmutadores, encaminadores o NIDS. La motivación alrededor de esta herramienta incluye: Emulación y simulación de ambientes de prueba similares a los encontrados en Internet. Puede generar tráfico más rico que otras aplicaciones como: SURGE, Iperf y Aspirent. Entre los principales beneficios que ofrece Harpoon se puede destacar los siguientes:

- Escalable, genera tráfico de red estadísticamente representativo a nivel de flujo IP.
- Independiente de las aplicaciones.
- Recrea volúmenes temporales característicos de flujo real.
- Recrea características espaciales como origen y destino.
- Se auto configura a partir de los registros de Netflow o de las trazas de paquetes.
- No se requiere estimación paramétrica.

8.3.8. Mace

MACE [97] es un ambiente de composición de tráfico malicioso, el cual recrea un amplio rango de tráfico de paquetes en un ambiente confinado de pruebas. Se define un modelo que permita una composición flexible de tráfico malicioso tanto de ataques conocidos como la experimentación con nuevos ataques. Basado en los análisis de fortalezas llevados a cabo

en detectores de intrusiones en redes de amplia difusión, se demostró las potenciales debilidades de estos sistemas a ataques simples lanzados simultáneamente de un pequeño número de servidores.

8.3.9. Nemean

Nemean [98] es un sistema para la generación automática de tráficos partir de la firmas de intrusiones obtenidas por honeynet. Su arquitectura se distingue por el marco de diseño modular y la semántica de conocimiento de protocolos, permitiendo la construcción de firmas que reduce notablemente las falsas alarmas. Los elementos constitutivos de la arquitectura incluyen la normalización de la capa de transporte y aplicación, el perfil de de agrupamiento y aprendizaje automatizado genera conexiones y sesiones de firmas significativas. Los experimentos han demostrado la utilidad de usar dos conjuntos de firmas para evaluar un sistema IDS:

- Un conjunto de datos para evaluar las falsas alarmas.
- Un conjunto honeynet para medir la relación de detecciones y firmas generadas

8.3.10. Honeycomb

Honeycomb [99] es un sistema desarrollado en la Universidad de Cambridge, el cual crea firmas de intrusiones usando honeypots. Automatiza la generación de firmas de ataques para NIDS aplicando técnicas de búsqueda de patrones y verifica la conformación de la cabecera de los paquetes, analizando protocolos en múltiples niveles de la jerarquía de estos. El propósito de usar honeypot es precisamente contar con tráfico malicioso ya que este es el fin de honeypod la captura de tráfico sospechoso.

8.3.11. Autograph

Autograph [100] se enfoca en los gusanos, los cuales se propagan en forma aleatoria buscando direcciones IP, basada en la búsqueda y clasificación del flujo de datos en los distintos

puertos. Este sistema genera en forma automática firmas de gusanos que se propagan en Internet usando la capa de transporte TCP, las firmas son generadas analizando la prevalencia del flujo de carga y por lo tanto no usa conocimiento de la semántica por encima del protocolo TCP. Es diseñado para producir firmas que exhiban alta sensibilidad (altos verdaderos positivos) y alta especificidad (bajos falsos positivos).

8.3.12. Polygraph

Es de amplio conocimientos que los sistemas IDS son fácilmente evadidos por los Gusanos Polimórficos, que varían su firma de ataque después de cada nuevo ataque de infección. Polygraph [101] es un sistema generador de firmas que reproduce el comportamiento de las firmas de los Gusanos Polimórficos. Polygraph genera firmas que consisten en múltiples cadenas disjuntas. Para realizar esto tiene en cuenta las funciones de exploit del mundo real, múltiples cadenas invariantes están en todas las variantes de las cargas, esas sub cadenas típicamente corresponden a marcos del protocolo, direcciones de retorno, y en algunos casos códigos pobremente ofuscados. La principal contribución a la definición de la generación de la firma polimórfica, proponiendo un conjunto de clases que apareen a la carga de los gusanos y presenten algoritmos para la generación automática de las firmas en esas clases. La evaluación de esas firmas en un rango de gusanos polimórficos ha demostrado que Polygraph produce firmas de gusanos polimórficos que exhiben bajos falsos negativos y falsos positivos.

8.4. Pruebas de Evasión

Las técnicas de evasión son modificaciones del tráfico para prevenir la detección por un sistema NIDS, los ataques de evasión constituyen un desafío fundamental para los NIDS, afectando su percepción básica del flujo de las redes. Si un NIDS interpreta el tráfico de red de una manera diferente que los puntos finales involucrados el NIDS no puede detectar el ataque eficientemente. El trabajo [102] describe el desarrollo de herramientas para evaluar en que grado los NIDS son vulnerables a distintas formas de evasión. La herramienta está automatizada y

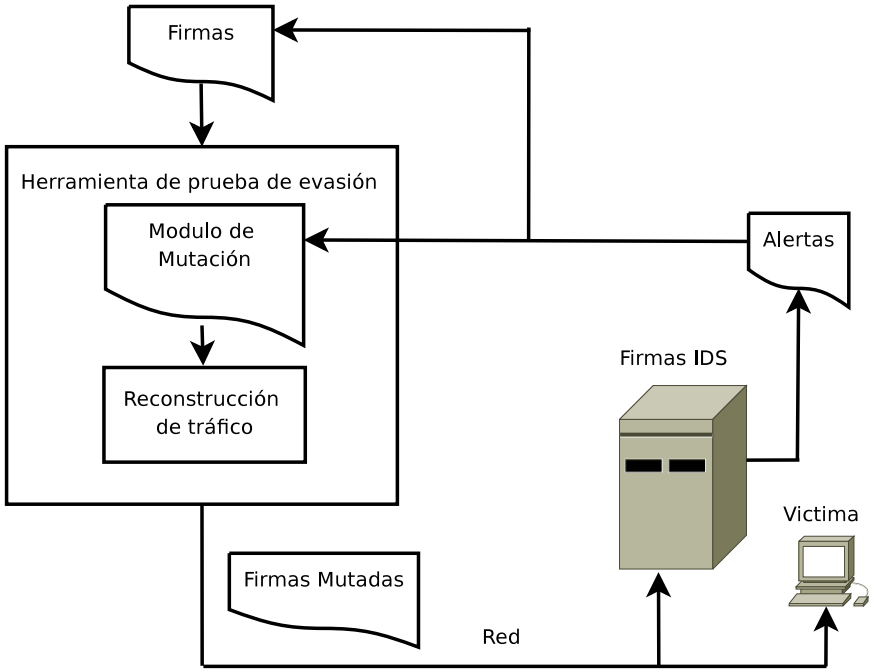

Figura 8.2: Marco de trabajo prueba de evasión.

posee una interfase gráfica amigable. En el trabajo de Yang se presenta una evaluación de la herramienta contra el programa NIDS Snort.

8.4.1. Técnicas de Evasión

Evasión o mutación de ataque [103] [104] [105] es un término que comprende una amplia variedad de técnicas para modificar el flujo de paquetes y que un ataque pueda ofuscar un NIDS y vulnerar los servicios. Por ejemplo la víctima puede aceptar paquetes que son ignorados por el NIDS, tales que el atacante envía paquetes extras con el mismo número de secuencia que los paquetes previos pero con diferentes contenidos en el campo de datos. El NIDS puede parar los paquetes porque el numero de secuencia ya fue usado, mientras que la víctima acepta y procesa los paquetes, reemplazando una cadena ya recibida con otra que produce la vulnerabilidad de la victima. Muchos otros tipos de ataque son posibles.
Como muestra la fig.8.2 las técnicas de evasión pueden proceder de la siguiente manera:

- Las firmas son colocadas en la máquina de mutación.
- La máquina de mutación modifica las firmas de acuerdo a los mecanismos de mutación.

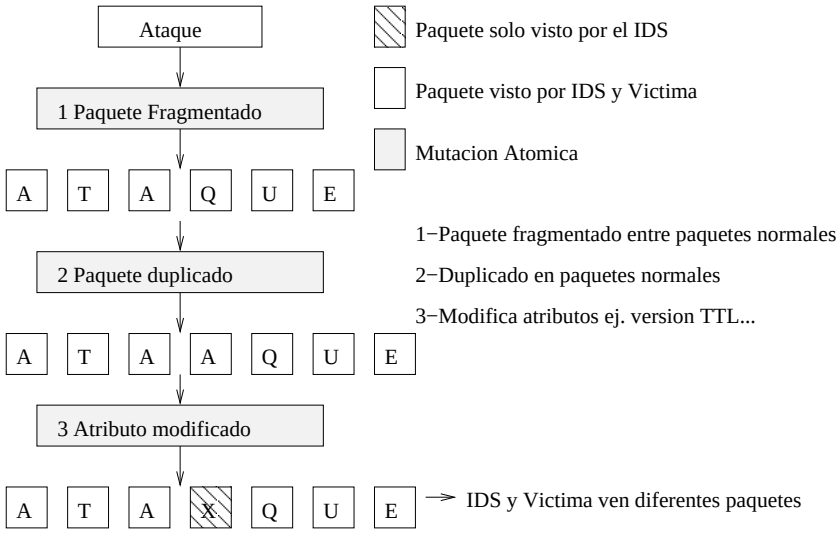

Figura 8.3: Ejemplo de evasión.

- La máquina de mutación envía el paquete mutado a la red.
- El paquete mutado evade la detección del NIDS y alcanza la máquina de la víctima.

Muchas técnicas de evasión son conocidas y operan a diferentes niveles del modelo OSI. una computadora víctima puede aceptar paquetes que rechaza el IDS. Un IDS que por error rechaza tales paquetes la computadora víctima pierde totalmente su contenido. Esta condición puede ser explotada, injectando subrepticiamente información crucial que pase el IDS en paquetes que el IDS es muy estricto para procesarlos. Esos paquetes evaden el escrutinio del IDS.
Un ejemplo de evasión IP es ilustrado en la fig 8.3. Un atacante confronta el NIDS con el flujo de paquetes en los cuales uno de los paquetes normales es duplicado y aceptado solo por el NIDS. Como resultado de esto el NIDS y la víctima reconstruyen diferentes flujos de datos.

Capítulo 9

Conclusiones

9.1. Logros de Investigación

La detección de intrusiones es una de las tareas más importantes y exigentes de la seguridad de las redes y el reconocimiento de patrones de ataques. Se ha abordado la detección de intrusiones con dos herramientas novedosas, las redes neuronales y las colonias de hormigas. En primer lugar, se comprueba que la red neuronal puede entrenarse fácilmente para distinguir entre condiciones normales y en condiciones de ataque de la red. Se obtuvo un error de aprendizaje de 0,56 % y un error de generalización de 1,97 %, utilizando una configuración sencilla de la red neuronal (configuración "back-propagation" con sólo dos neuronas en la capa oculta). Este resultado se ha logrado por el uso del módulo procesador estadístico que colabora en forma eficiente en la detección de intrusiones. Como se puede advertir, para la detección de inundaciones UDP el aporte realizado por los parámetros relacionados con el protocolo IP es pequeño en relación al aporte de los parámetros del protocolo UDP al momento de llevar a cabo la detección. Por esta razón, se podría omitir la utilización de los parámetros del protocolo IP. El sistema se podría adaptar para detectar otros tipos de intrusiones, tales como inundaciones TCP-SYN e ICMP (Internet Control Message Protocol, Protocolo Internet de Mensajes de Control). En el caso de inundaciones TCP-SYN, se debería incluir el uso de parámetros relacionados al protocolo TCP mientras que las inundaciones de paquetes ICMP se detectarían a

través de los parámetros del protocolo IP. Esto será factible siempre que la red neuronal pueda aprender de los distintos tipos de ataque. Se debe tener en cuenta que, en este caso, no se tiene control sobre el tráfico de fondo de la red. Por el contrario si se tuviese control de este, a través de programas de simulación y de modelado de tráfico de red, se podría analizar el rendimiento del sistema para distintos tráficos de fondo y distintos niveles de ataque para mejorar su desempeño. Para ataques más complejos no se ha logrado tener un buen desempeño con este método por lo que se optó por el método de colonias de hormigas. Con la investigación en la optimización de colonias de hormigas, se ha propuesto un enfoque en el aprendizaje supervisado ACCM para detección de intrusos y anomalías en redes. La ACCM mejora los algoritmos de agrupamiento existentes basado en colonia de hormigas buscando que la agrupación sea casi óptima. Determina automáticamente el número de conglomerados que es necesario para ser introducidas en otros algoritmos de agrupamiento como K-Means, EM y el agrupamiento Fuzzy c-means. Además, ACCM es insensible a los valores extremos, que existen habitualmente en el tráfico de red y corrompen la estructura del agrupamiento. Los resultados experimentales demuestran que, comparando con el agrupamiento de hormigas existentes basados en algoritmo, ACCM reduce significativamente el número de agrupaciones estadísticamente equivalentes, y genera una agrupación cercana a la solución óptima . Los resultados también indican que la aplicación de la ACCM extendido Infomax, algoritmo ICA es eficaz para detectar los ataques conocidos y las intrusiones no vistas de alta tasa de detección y reconocer el tráfico de red normal con baja tasa de falsos positivos. Por otra parte, se ha abordado dos cuestiones importantes para la detección de intrusos supervisado:

- Generación de sistemas difusos exactos e interpretable para la clasificación.
- Técnicas de evaluación de la función de selección para el dominio de detección de intrusos.

Generación automática de conocimiento basado en normas y minería de datos ha sido ampliamente adoptado por su precisión de clasificación. Sin embargo, la atención no se ha centrado en la optimización sistemas de reglas basadas en normas, que es importante para el análisis de

intrusiones. Se propone un novedoso sistema de detección de intrusos - MOGFIDS para extraer en forma exacta e interpretable el conocimiento basado en reglas difusas a partir de datos de tráfico de red utilizando un marco evolutivo basado en agentes. Los ataques crecen en complejidad y sofisticación, adoptando formas polimórficas, esto demanda cada vez más IDS con mejores rendimientos y más inteligentes. A partir de la colonia de hormigas se ha presentado un sistema de auto aprendizaje no supervisado el ACCM para la detección de intrusiones en redes. El ACCM mejora los algoritmos de colonias de hormigas existentes buscando una heurística de agrupamiento cercana a la óptima. Se han introducido conceptos tales como entropía regional local, infraestructura de feromonas, memoria modificada a corto termino, selección de corridas, balance entre exploración y explotación, optimización entre unión y separación de agrupamientos. El ACCM determina automáticamente el número de agrupaciones crítica requerida como entrada para los algoritmos K-Means y E-M. El ACCM es insensible para los valores extremos, los cuales existen en el tráfico de redes y corrompen la estructura de agrupamientos.

Los resultados de las pruebas demuestran que en comparación con los algoritmos de hormigas existentes, el ACCM reduce significativamente el número de agrupaciones estadísticamente equivalentes y genera una solución de agrupamientos cercana a la óptima. En general la calidad de agrupaciones y el rendimiento de clasificación del ACCM es respectivamente más estable y preciso que K-Means y E-M, como se mostró en las tablas. Los resultados obtenidos demuestran que la aplicación del ACCM junto con el algoritmo Ext-InfomaxICA, es efectiva para detectar ataques conocidos y nuevos en redes, con una relación de detección alta y con reconocimiento de tráfico normal con una baja tasa de falsos positivos. Finalmente se propone una arquitectura de multi agentes distribuidos basados en el ACCM para detectar intrusiones en grandes redes.

9.2. Futuros trabajos

En cuanto al enfoque de detección de anomalía no supervisadas, la calidad de la agrupaciones y precisión en la clasificación del ACCM propuesto ha sido demostrado por algunos en el

mundo real, con el conjuntos de datos de referencia y Cup99 KDD-IDS. En un trabajo futuro, se planea mejorar el rendimiento en tiempo de ACCM para agrupaciones más grandes y datos de mayor dimensión. Además, abrirá una investigación sobre la hibridación de ACCM y otros algoritmos de agrupamiento. Esta propuesta, principalmente explora si ACCM puede cooperar con los algoritmos de búsqueda de agrupamientos locales para extraer más estructuras óptimas de agrupamiento utilizando diferentes funciones objetivo en los diferentes niveles de agrupación. En orden para desarrollar un inteligente y escalable IDS, la grandes redes de conmutación de hoy en día, las múltiples agrupación de colonias de hormigas heterogéneas, el enfoque será diseñado para integrarse en el desarrollo de la arquitectura multi-agente IDS ACCM. Teniendo en cuenta el enfoque de supervisión de detección de intrusos, los resultados experimentales llevados a cabo han demostrado con éxito la clasificación de los ataques de intrusos y el tráfico de red normal, por lo tanto, hay mucho margen para la labor futuras aplicaciones de este enfoque a otros dominios de problemas complejos como el reconocimiento facial y la informática de ADN, que puede ser estudiada con sistemas difusos exacta e interpretable.

Bibliografía

[1] J. P. Anderson, "Computer security technology planing study," *ESD-TR-73-51 Electronics System Division(AFSC)*, vol. 1, pp. 1–32, october 1972.

[2] *DEPARTMENT OF DEFENSE TRUSTED COMPUTER SYSTEM EVALUATION CRITERIA*, DoD Std. DoD 5200.28-STD, 1985.

[3] *Introduction and overview for the ISMS Family of Standards, plus a glossary of common terms*, International Organization for Standardization Std. ISO27 000, 2008.

[4] *Data Comunication Networks: Open system interconnection OSI; Security Structure and Applications*, INTERNATIONAL TELECOMMUNICATION UNION Std. X800, 1991.

[5] *Security architecture for Systems providing end-to-end communications*, INTERNATIONAL TELECOMMUNICATION UNION Std. X805, 2003.

[6] *Internet Security Glossary*, Network Working Group The Internet Society Std. RFC 2828, 2000.

[7] S. F. Kenneth Ingham, "A history and survey of network firewalls," *ACM*, vol. V, no. N, pp. 1–42, 2002.

[8] R. F. R. M. J. A. Jeffrey C. Mogul, "The packet filter: An efficient mechanism for user-level network code," *Digital Western Reserch Laboratory*, pp. 1–34, 1987.

[9] T. H. Ptacek and T. N. Newsham, “Insertion, evasion, and denial of service: Eluding network intrusion detection,” Secure Networks, Inc., Suite 330, 1201 5th Street S.W, Calgary, Alberta, Canada, T2R-0Y6, Tech. Rep., 1998.

[10] J. P. Anderson, “Computer security threat monitoring and surveillance.” *Technical report J. P. Anderson Co*, april 1980.

[11] C. G. Martin Roesch. (2007) Snort homepage on. http://www.snort.org/docs/snort_htmanuals/htmanual_280/node1.html.

[12] V. Paxson. (2007) BRO-IDS homepage on. http://www.bro-ids.org/Features.html.

[13] C. Manikopoulos and S. Papavassiliou, “Network intrusion and fault detection: a statistical anomaly approach,” *Communications Magazine*, vol. 40, pp. 76–82, Oct. 2002.

[14] T. Ekola, M. Laurikkala, T. Lehto, and H. Koivisto, “Network traffic analysis using clustering ants,” *World Automation Congress, 2004. Proceedings*, vol. 17, pp. 275–280, 28 June - 1 July 2004.

[15] H.-H. Gao, H.-H. Yang, and X.-Y. Wang, “Ant colony optimization based network intrusion feature selection and detection,” *Machine Learning and Cybernetics, 2005. Proceedings of 2005 International Conference on*, vol. 6, pp. 3871–3875 Vol. 6, 18-21 Aug. 2005.

[16] R. Gopalakrishna, “A framework for distributed intrusion detection using interest-driven cooperating agents,” Technical Report 2001-44, CERIAS, Department of Computer Science,Purdue University, Lafayette, Indiana, Tech. Rep., 20011998.

[17] R. J. Anderson, *Security Engineering: A Guide to Building Dependable Distributed Systems*. New York, NY, USA: John Wiley & Sons, Inc., 2001.

[18] N. S. A. S. S. F. G. G. M. MD, “National information systems security (infosec) glossary,” *NATIONAL SECURITY AGENCY/CENTRAL SECURITY SERVICE FORT GEORGE G MEADE MD*, no. A929334, pp. 1–81, september 2000.

[19] M. B. Magno, *Survey of User Authentication Mechanisms*. MONTEREY CA: NAVAL POSTGRADUATE SCHOOL MONTEREY CA, 1996.

[20] M. Sánchez, B. Jiménez, and F. Gutiérrez, "Estudio del control de acceso en sistemas colaborativos," *XII JORNADAS DE INGENIERIA DEL SOFTWARE Y BASES DE DATOS Y II CONGRESO ESPAÑOL DE INFORMATICA*, vol. Volume 1, pp. 55 – 62, December 2007.

[21] D. Mellado, E. Fernandez-Medina, and M. Piattini, "A security requirements engineering process in practice," *Latin America Transactions, IEEE (Revista IEEE America Latina)*, vol. 5, no. 4, pp. 211–217, July 2007.

[22] R. Sandhu, E. Coyne, H. Feinstein, and C. Youman, "Role-based access control: a multi-dimensional view," *Computer Security Applications Conference, 1994. Proceedings., 10th Annual*, pp. 54–62, Dec 1994.

[23] R. Sandhu, E. Coyne, H. Feinstein, and C. Youman, "Role-based access control models," *Computer*, vol. 29, no. 2, pp. 38–47, Feb 1996.

[24] R. Thomas and R. Sandhu, "Conceptual foundations for a model of task-based authorizations," *Computer Security Foundations Workshop VII, 1994. CSFW 7. Proceedings*, pp. 66–79, Jun 1994.

[25] G. Stoneburner, *Underlying Technical Models for Information Technology Security*. Computer Security Division Information Technology Laboratory National Institute of Standards and Technology Gaithersburg, MD 20899-8930: NIST Special Publication 800-33, 2001.

[26] *ISO. Risk management - vocabulary - guidelines for use in standards*, ISO/IEC International Organization for Standardization Std. ISO/IEC Guide 73, 2002.

[27] B. Schneier, *Security Engineering*. New York, NY: John Wiley, 2004.

[28] A. S. Quist. (2007) Security Classification of Information homepage on. http://www.fas.org/sgp/library/quist/index.html.

[29] M. McDowell. (2007) National Cyber Alert System Cyber Security Tip ST04-015 Understanding Denial-of-Service Attacks homepage on. http://www.us-cert.gov/cas/tips/ST04-015.html.

[30] CERT. (2007) CERT Coordination Center Denial of Service Attacks homepage on. http://www.cert.org/tech_tips/denial_of_service.html.

[31] CERT. (1998) CERT Advisory CA-1998-01 Smurf IP Denial-of-Service Attacks homepage on. http://www.cert.org/advisories/CA-1998-01.html.

[32] S. Kumar, "Smurf-based distributed denial of service (ddos) attack amplification in internet," *Internet Monitoring and Protection, 2007. ICIMP 2007. Second International Conference on*, pp. 25–25, 1-5 July 2007.

[33] S. Kumar, M. Azad, O. Gomez, and R. Valdez, "Can microsoft's service pack2 (sp2) security software prevent smurf attacks?" *Telecommunications, 2006. AICT-ICIW 06. International Conference on Internet and Web Applications and Services/Advanced International Conference on*, pp. 89–89, 19-25 Feb. 2006.

[34] W. M. Eddy, "Defenses against tcp syn flooding attacks," *The Internet Protocol Journal*, vol. 9-4, december 2006.

[35] CERT. (1997) CERT Advisory CA-1997-28 IP Denial-of-Service Attacks homepage on. http://www.cert.org/advisories/CA-1997-28.html.

[36] C. P. Alefiya Hussain, John Heidemann, "A framework for classifying denial of service attacks," *SIGCOMM 2003 conference*, pp. 99–110, August 2003.

[37] S. de Vries, "Application denial of service (dos) attacks," 2004, http//:research.corsaire.com/whitepapers/040405-application-level-dos-attacks.pdf.

[38] Wikipedia. (2008) Denial-of-service attack homepage on. http://en.wikipedia.org/wiki/Denial-of-service_attack.

[39] T. F. Nong Ye, Bashettihalli Harish, "Attack profiles to derive data observations, features, and characteristics of cyber attacks," *Information, Knowledge, Systems Management*, vol. 5, no. 1, pp. 1–25, 2006.

[40] T. A. L. John D. Howard, "A common language for computer security incidents," *Sandia Report*, pp. 23–47, 1998.

[41] T. A. L. John D. Howard. (1997) An Analysis of Security Incidents on the Internet Ph.D. Dissertation, Department of Engineering and Public Policy, Carnegie Mellon University, Pittsburgh, PA, April, 1997. homepage on. http://www.cert.org/.

[42] Y. Bai and H. Kobayashi, "Intrusion detection systems: technology and development," *Advanced Information Networking and Applications, 2003. AINA 2003. 17th International Conference on*, vol. Issue , 27-29, pp. 710 – 715, March 2003.

[43] G. Vigna and A. Kemmerer, "Netstat: a network-based intrusion detection approach," in *Computer Security Applications Conference*, vol. 1, Dec. 1998, pp. 25–34.

[44] A. Valdes and D. Anderson, "Statistical methods for computer usage anomaly detection using nides," *Tech. rep.: SRI International*, Jan 1995.

[45] V. Paxson, "Bro: A system for detecting network intruders in real-timeinternet," *In Proceedings of the 7th USENIX Security Symposium, San Antonio*, pp. 1–18, Juanary 1998.

[46] E. H. S. Rajeev Gopalakrishna, "A framework for distributed intrusion detection using interest-driven cooperating agents," *Technical Report 2001-44, CERIAS, Department of Computer Science, Purdue University*, pp. 1–23, 2001.

[47] J. McHugh, A. Christie, and J. Allen, "The role of intrusion detection systems," *IEEE Software*, pp. 42 – 52, September-October 2000.

[48] M. Almgren and U. Lindqvist, "Application-integrated data collection for security monitoring," in *Recent Advances in Intrusion Detection (RAID 2001)*, ser. LNCS. Davis, California: Springer, October 2001, pp. 22–36, /urlhttp://www.sdl.sri.com/papers/raid2001/.

[49] U. Lindqvist and P. A. Porras, "Detecting computer and network misuse through the production-based expert system toolset (p-best)," in *Proceedings of the 1999 IEEE Symposium on Security and Privacy*. Oakland, California: IEEE Computer Society Press, Los Alamitos, California, may 1999, pp. 146–161, http://www.csl.sri.com/papers/pbest-sp99-cr/.

[50] T. Champion and M. Denz, "A benchmark evaluation of network intrusion detection systems," *Aerospace Conference, 2001, IEEE Proceedings.*, vol. 6, pp. 2705–2712 vol.6, 2001.

[51] IMB. (2008) IBM Proventia^{TM}Content Analyzer Technology Hybrid Protection (System & Data Security) for IBM Proventia Network Intrusion Prevention System homepage on. http://www-935.ibm.com/services/us/iss/pdf/wp_ibm-proventia-content-analyzer-technology.pdf.

[52] N. Labs. (2008) IBM PROVENTIA NETWORK IPS GX6116 PRODUCT CERTIFICATION homepage on. http://nsslabs.com/test\-reports/NSS\-NIPS\-IBM\-GX6116.pdf.

[53] N. Group. (2006) NSS Group homepage on. http://nssgroup.org/grouptests/ips/edition3/cisco4255/cisco4255.htm.

[54] N. Labs. (2008) Cisco IPS-4255 V5.0(3)Technical Evaluation homepage on. http://nsslabs.com/grouptests/ips/edition3/pdf/Cisco20IPS\-425520V5.02832920WP.pdf.

[55] N. Labs. (2008) MCAFEE INTRUSHIELD 4010 V3.1.3 Technical Evaluation homepage on. http://nsslabs.com/certification/mgips/test\-reports/MGIPS\-0610\-MCA.pdf.

[56] N. Labs. (2008) SecureWorks iSensor 850 Technical Evaluation homepage on. http://nsslabs.com/grouptests/ips/edition3/pdf/IPSED3\-0601\-SW.pdf.

[57] N. Labs. (2008) JUNIPER NETWORKS IDP 600F V3.1 Technical Evaluation homepage on. http://nsslabs.com/grouptests/ips/edition3/pdf/Juniper20Networks20IDP20600F20V3120WP.pdf.

[58] C. Point. (2007) Check Point homepage on. http://www.checkpoint.com/products/ips\-1/index.html.

[59] radware. (2007) DefensePro homepage on. http://www.radware.com/Products/ApplicationNetworkSecurity/DefensePro.aspx.

[60] DeepNines. (2008) DeepNines Technologies homepage on. http://www.deepnines.com/products/Intrusion_Prevention.php.

[61] N. S. Foundation. (2008) Bro Intrusion Detection System homepage on. http://www.bro-ids.orgf.

[62] Gartner. (2008) Magic Quadrant for Network Intrusion Prevention System Appliances homepage on. http://mediaproducts.gartner.com/reprints/tippingpoint/154849.html.

[63] G. Fink, B. Chappell, T. Turner, and K. O'Donoghue, "A metrics-based approach to intrusion detection system evaluation for distributed real-time systems," *Parallel and Distributed Processing Symposium., Proceedings International, IPDPS 2002, Abstracts and CD-ROM*, pp. 93–100, 2002.

[64] N. Labs. (2008) NSS Labs homepage on. http://www.nsslabs.com/.

[65] J. A. B. D. Cabrera, L. Lewis, X. Qin, W. Lee, R. K. Prasanth, B. Ravichandran, and R. K. Mehra, "Proactive detection of distributed denial of service attacks using mib traffic variables a feasibility study," *Proceedings of the 7th IFIP/IEEE International Symposium on Integrated Network Management, WA - May 14-18,2001*, 2001.

[66] C. Mellon. (2007) Statistical-Based Intrusion Detection homepage on. http://www.sei.cmu.edu/str/descriptions/sbid_body.html.

[67] X. Jin and R. Huang, "Detecting network attacks via improved iterative scaling," *Industrial Informatics, 2007 5th IEEE International Conference on*, vol. 1, pp. 113–118, June 2007.

[68] S. A. Alvarez, "Chi-squared computation for association rules:preliminary results," *Technical Report BC-CS-2003-01*, 2003.

[69] Y. Nong *et al.*, "Statistical process control for computer intrusion detection," *DARPA Information Survivability Conference Exposition II, 2001. DISCEX '01. Proceedings*, vol. Volume 1, pp. 3 – 14, June 2001.

[70] Wikipedia. (2008) Wikipedia homepage on. http://en.wikipedia.org/wiki/Artificial_neural_network.

[71] C. M. Bishop, *Neural Networks for Pattern Recognition*. New York, NY: Oxford Univ. Press, 1995.

[72] P. Ramasubramanian and A. Kannan, "Intelligent multi-agent based back-propagation neural network forecasting model for statistical database anomaly prevention system," *Intelligent Sensing and Information Processing, 2004. Proceedings of International Conference on*, pp. 108 – 113, 2004.

[73] S. C. Lee and D. Heinbuch, "Training a neural-network based intrusion detector to recognize novel attacks," *Systems, Man and Cybernetics, Part A, IEEE Transactions on*, vol. Volume 31, pp. 294 – 299, July 2001.

[74] L. G. Allred and G. E. Kelly, "Supervised learning techniques for backpropagation networks," *Neural Networks 1990., 1990 IJCNN International Joint Conference on*, vol. Volume 1, pp. 721 – 728, 17-21 June 1990.

[75] P.-C. Lin, Y.-D. Lin, Y.-C. Lai, and T.-H. Lee, "Using string matching for deep packet inspection," *Computer*, vol. 41, no. 4, pp. 23–28, April 2008.

[76] G. Navarro and M. Raffinot, "New techniques for regular expression searching," *Algorithmica*, vol. 41, no. 2, pp. 89–116, 2004, http://citeseer.ist.psu.edu/699874.html.

[77] B. H. Bloom, "Space/time trade-offs in hash coding with allowable errors," *Commun. ACM*, vol. 13, no. 7, pp. 422–426, 1970.

[78] I. Sourdis, a. B. Jo a. M. C. Jo and S. Vassiliadis, "Regular expression matching in reconfigurable hardware," *J. Signal Process. Syst.*, vol. 51, no. 1, pp. 99–121, 2008.

[79] V. Paxson, K. Asanović, S. Dharmapurikar, J. Lockwood, R. Pang, R. Sommer, and N. Weaver, "Rethinking hardware support for network analysis and intrusion prevention," in *HOTSEC'06: Proceedings of the 1st conference on USENIX Workshop on Hot Topics in Security*. Berkeley, CA, USA: USENIX Association, 2006, pp. 11–11.

[80] E. D. Lumer and B. Faieta, "Diversity and adaptation in populations of clustering ants," in *SAB94: Proceedings of the third international conference on Simulation of adaptive behavior : from animals to animats 3*. Cambridge, MA, USA: MIT Press, 1994, pp. 501–508.

[81] B. M. Julia Handl, Joshua Knowles and M. Dorigo, "Strategies for the increased robustness of ant-based clustering," *Engineering Self-Organising Systems, LNCS 2977*, pp. 90–104, 2004.

[82] A. Asuncion and D. Newman. (2007) UCI machine learning repository. [Online]. Available: http://www.ics.uci.edu/sim/mlearn/Repository.html

[83] C. E. Shannon and W. Weaver, *The Mathematical Theory of Communication*. Urbana, Illinois: University of Illinois Press, 1949.

[84] G. Theraulaz, J. Gautrais, S. Camazine, and J. L. Deneubourg, "The formation of spatial patterns in social insects: from simple behaviours to complex structures." *Philos*

Transact A Math Phys Eng Sci, vol. 361, no. 1807, pp. 1263–1282, June 2003. [Online]. Available: http://dx.doi.org/10.1098/rsta.2003.1198

[85] J. A. Sauter, R. Matthews, H. Van Dyke Parunak, and S. Brueckner, "Evolving adaptive pheromone path planning mechanisms," in *AAMAS '02: Proceedings of the first international joint conference on Autonomous agents and multiagent systems*. New York, NY, USA: ACM, 2002, pp. 434–440.

[86] K. V. Mardia, J. M. Bibby, and J. T. Kent, *Multivariate analysis / K. V. Mardia, J. T. Kent, J. M. Bibby*. Academic Press, London ; New York :, 1979. [Online]. Available: http://www.loc.gov/catdir/toc/els031/79040922.html

[87] X. Yu, X. Cheng, Y. Fu, J. Zhou, H. Hao, X. Yang, H. Huang, T. Zhang, and L. Fang, "Research of independent component analysis," vol. 5, 0-0 2004, pp. 4804 –4809 vol.5.

[88] A. J. Bell and T. J. Sejnowski, "An information-maximization approach to blind separation and blind deconvolution," *Neural Computation*, vol. 7, no. 6, pp. 1129–1159, 1995. [Online]. Available: http://www.mitpressjournals.org/doi/abs/10.1162/neco.1995.7.6.1129

[89] T.-W. Lee, M. Girolami, and T. J. Sejnowski, "Independent component analysis using an extended infomax algorithm for mixed subgaussian and supergaussian sources," *Neural Computation*, vol. 11, no. 2, pp. 417–441, 1999. [Online]. Available: http://www.mitpressjournals.org/doi/abs/10.1162/089976699300016719

[90] A. Hyvarinen, "Fast and robust fixed-point algorithms for independent component analysis," *Neural Networks, IEEE Transactions on*, vol. 10, no. 3, pp. 626 –634, may 1999.

[91] B. Systems. (2009) Breakingpoint Systems homepage on. http://www.breakingpointsystems.com/products/breakingpointelite.

[92] C. Tecnologies. (2009) CANDELA Tecnologies homepage on. http://www.candelatech.com/.

[93] I. S. I. USC. (2008) Traffic Generator homepage on. http://www.postel.org/tg/.

[94] K. V. Vishwanath and A. Vahdat, "Realistic and responsive network traffic generation," in *SIGCOMM '06: Proceedings of the 2006 conference on Applications, technologies, architectures, and protocols for computer communications*. New York, NY, USA: ACM, 2006, pp. 111–122.

[95] S. Singh, C. Estan, G. Varghese, and S. Savage, "Automated worm fingerprinting," *6th Symposium on Operatim System Design and Implementation*, pp. 28–38, 2004.

[96] J. Sommers, V. Yegneswaran, and PaulBarford, "Toward comprehensive traffic generation for online ids evaluation," *Tecnical Report N 1525 Computer Sciences Dept., University Wisconsin-Madison*, 2005.

[97] J. Sommers, V. Yegneswaran, and PaulBarford, "A framework for malicious worload generation," in *IMC '04: Proceedings of the 4th ACM SIGCOMM conference on Internet measurement*. New York, NY, USA: ACM, 2004.

[98] V. Yegneswaran, J. T. Giffin, P. Barford, and S. Jha, "An architecture for generating semantics-aware signatures," in *SSYM'05: Proceedings of the 14th conference on USENIX Security Symposium*. Berkeley, CA, USA: USENIX Association, 2005, pp. 7–7.

[99] C. Kreibich and J. Crowcroft, "Honeycomb: creating intrusion detection signatures using honeypots," *SIGCOMM Comput. Commun. Rev.*, vol. 34, no. 1, pp. 51–56, 2004.

[100] H.-A. Kim and B. Karp, "Autograph: Toward automated, distributed worm signature detection," *Proceedings of the 13th USENIX Security Symposium*, pp. 271–286, 2004.

[101] J. Newsome, B. Karp, and D. Song, "Polygraph: automatically generating signatures for polymorphic worms," in *Security and Privacy, 2005 IEEE Symposium on*, May 2005, pp. 226–241.

[102] C.-H. Yang, C.-H. Lin, L.-C. Juan, J.-S. Wu, and T.-Y. Huang, "Design and development of an nids evasion test tool with gui," in *The 2nd Workshop on Information Security*. Waseda Tokyo, Japan: JWIS Association, 2007, pp. 1–8.

[103] T. Ptacek, T. Newsham, and H. J. Simpson, "Insertion, evasion, and denial of service: Eluding network intrusion detection," Tech. Rep., 1998.

[104] N. J. Puketza, K. Zhang, M. Chung, Y. Chung, B. Mukherjee, and R. A. Olsson, "A methodology for testing intrusion detection systems," *IEEE Transactions on Software Engineering*, vol. 22, pp. 719–729, 1996.

[105] U. Shankar and V. Paxson, "Active mapping: resisting nids evasion without altering traffic," in *Security and Privacy, 2003. Proceedings. 2003 Symposium on*, May 2003, pp. 44–61.