Revista Cubana de Ciencias Informáticas
ISSN: 2227-1899 | RNPS: 2301
Vol. 8, No. 4, Octubre-Diciembre, 2014
Págs. 52-73
http://rcci.uci.cu

Tipo de artículo: Artículo de revisión
Temática: Seguridad informática
Recibido: 15/01/2014 | Aceptado: 2/07/2014

Técnicas de aprendizaje automático para la detección de intrusos en redes de computadoras

Machine learning techniques for intrusion detection in computer networks

Jorge Luis Rivero Pérez

Universidad de Cienfuegos "Carlos Rafael Rodríguez". Carretera a Rodas, km 4. Cienfuegos, Cuba. Correo-e: jlrivero@ucf.edu.cu

Resumen

El desarrollo de sistemas de detección de intrusos en redes de computadoras (del inglés NIDS) constituye un reto para los investigadores, debido a que con el crecimiento de las redes de computadoras, aparecen, constantemente nuevos ataques basados en contenido. El presente artículo además de hacer una descripción de los enfoques de detección de intrusos basados en firmas y en anomalías, constituye una revisión de las diferentes técnicas de aprendizaje automático a aplicar en las etapas de preprocesamiento y procesamiento de los datos para la detección. Se describen la taxonomía de los NIDS y un esquema de clasificación de atributos de conexiones. En la detección de anomalías a partir de técnicas de aprendizaje automático varios son los conjuntos de datos empleados, siendo KDD Cup 99 el más utilizado. Atendiendo a esto se describe ese conjunto de datos y se exponen resultados obtenidos sobre el mismo a partir de algunas técnicas de preprocesamiento de datos como selección y discretización. Son expuestos novedosos enfoques que hibridan algoritmos de búsqueda basados en inteligencia de enjambre con algoritmos de aprendizaje automático, lo que posibilita elevar los índices de detección y mejoran la detección de ataques basados en contenido. Esta revisión resulta de gran aplicabilidad a investigadores que buscan áreas dentro de la detección de intrusos en redes de computadoras a partir de técnicas de aprendizaje automático, donde realizar aportes, por lo que se recomienda su consulta.

Palabras clave: aprendizaje automático, detección de intrusos, inteligencia de enjambre, KDD Cup 99.

Revista Cubana de Ciencias Informáticas
ISSN: 2227-1899 | RNPS: 2301
Vol. 8, No. 4, Octubre-Diciembre, 2014
Págs. 52-73
http://rcci.uci.cu

Abstract

The development of network intrusion detection systems (NIDS) is a challenge for researchers, due to the growth of computer networks, constantly appear new content-based attacks. This article in addition to do a description of the approaches of intrusion signature-based and anomaly detection ones, also constitutes a review of the different machine learning techniques for the intrusion detection to be applied in data preprocessing and processing stages. NIDS taxonomy and an attributes classification scheme are described. In anomaly detection from machine learning techniques several data sets are employed, being KDD Cup 99 the most used. That data set is described and the results of some data preprocessing techniques applied on it such as selection and discretization are presented. Novel approaches that use search algorithms based on swarm intelligence with machine learning algorithms are exposed, which increase detection rates and improve the detection of content-based attacks. This review is of great relevance to researchers looking for areas within the intrusion detection in computer networks using machine learning techniques, in which make contributions.

Keywords: *intrusion detection, KDD Cup 99, machine learning, swarm intelligence.*

Introducción

Con el crecimiento de las redes de computadoras, el aumento de los servicios que ofrecen las mismas y la necesidad de mantener la confiabilidad, integridad y disponibilidad de la información transmitida, hace que la seguridad de los sistemas de cómputo gane más importancia, debido a que por otra parte aumentan los ataques a sistemas, convirtiéndose en un serio problema. Tal afirmación se puede constatar con el Informe Anual de Seguridad 2014 de Cisco (Cisco, 2014), donde destacan el crecimiento alarmante de vulnerabilidades, el mayor desde el año 2000, aprovechando nuevos frentes de ataque y técnicas renovadas. El informe destaca además la merma de la capacidad de las organizaciones para monitorizar y blindar sus redes. Además, el 100 por ciento de una muestra de 30 de las mayores redes corporativas del mundo generó tráfico hacia sitios web que albergan malware y el 96 por ciento de las redes analizadas dirigió tráfico hacia servidores "secuestrados", mientras el 92 por ciento transmitió tráfico a páginas web sin contenido, que típicamente albergan actividad maliciosa. Los ataques de denegación de servicio distribuidos (DDoS) que afectan al tráfico dirigido o generado desde sitios web atacados y pueden paralizar los proveedores de servicios de Internet, han aumentado tanto en volumen como en gravedad y los troyanos multipropósito constituyen la amenaza web más frecuentemente encontrada, representando el 27 por ciento del total de amenazas detectadas en 2013. Los ataques sencillos que causaban daños controlables han dado paso a operaciones ciber-criminales organizadas más sofisticadas, financiadas y capaces de causar un importante daño económico y de reputación tanto

para organizaciones públicas como privadas, atentando así contra la Seguridad Nacional de cualquier país. Además existe una mayor complejidad de las amenazas y de las soluciones debido al crecimiento exponencial de dispositivos móviles y entornos Cloud. Las nuevas clases de dispositivos inteligentes y las nuevas infraestructuras han ampliado el campo de acción de los atacantes, quienes pueden aprovecharse de las vulnerabilidades imprevistas y de sistemas de defensa inadecuados. Los ciber-criminales han aprendido que aprovechar el poder de la infraestructura de Internet les proporciona muchos más beneficios que el simple acceso a ordenadores o dispositivos individuales. Estos ataques a nivel de infraestructura pretenden obtener acceso a los servidores clave que albergan las páginas webs, servidores de nombres y data centers, con el fin último de extender las amenazas a innumerables activos individuales que se apoyan en estos recursos. Al atacar la infraestructura de Internet, los ciber-delincuentes debilitan la confianza en todo aquello que depende de dicha infraestructura (Cisco, 2014).

Los ataques son protagonizados por personas denominadas intrusos. Hay dos tipos de intrusos: intrusos externos, siendo usuarios no autorizados en los sistemas de cómputo que atacan y los intrusos internos, que tienen acceso restringido a los sistemas. Por tal motivo se hace necesaria una línea de defensa para proteger los sistemas ante los ataques, apareciendo así la detección de intrusos, sirviendo como una pared adicional que permite detectar ataques a los sistemas (Dong, *et al.*, 2013). En (Heady *et al.*, 1990) se define intrusión como "cualquier acción que atente y comprometa la integridad, confidencialidad o disponibilidad de un recurso" La solución es el uso de sistemas de detección de intrusos (IDS) que inspeccionan la actividad de los sistemas de cómputo en busca de patrones o de comportamiento considerado anormal que puede indicar un ataque al sistema o un mal uso. Existen dos categorías principales de técnicas para la detección de intrusos: Detección de anomalías y detección a partir del mal uso o basada en firmas (Lippmann *et al.*, 2000; Sung and Mukkamala, 2003). Investigaciones recientes exponen un crecimiento de ataques no conocidos basados en contenido, por lo que se hace necesario nuevas técnicas de preprocesamiento de datos así como mejorar los índices de detección de anomalías (Kaur *et al.*, 2013; Patel *et al.*, 2013). En este trabajo se realiza un estudio de detección de anomalías a partir de técnicas de aprendizaje automático, e hibridaciones de las mismas con otras técnicas de Inteligencia Artificial, como inteligencia de enjambre, identificando los pasos claves como son el preprocesamiento de los datos para reducción de dimensionalidad y las técnicas de detección. El objetivo es hacer un estudio del estado del arte que permita descubrir cuestiones abiertas en las etapas de preprocesamiento y procesamiento de los datos con un enfoque de aprendizaje automático.

Métodos

La investigación fue realizada a partir de la revisión de numerosos artículos relacionados con la detección de intrusos bajo un enfoque de aprendizaje automático, determinando así posibles cuestiones abiertas en esa área, donde se pudiera profundizar y hacer aportes. La investigación se centra en una revisión de las etapas de preprocesamiento de los datos, en el conjunto de datos más empleado en esta área y en el procesamiento semi-supervisado de los datos. En la selección de los métodos se tuvieron en cuenta aspectos como:

- Los datos que se necesitaba obtener.
- Correspondencia con el diseño teórico.
- Estrategia investigativa seleccionada.

Se logró avanzar en el proceso de investigación haciendo uso de métodos de trabajo científico como:

Métodos generales: El método hipotético-deductivo para proponer líneas de trabajo a partir de resultados parciales; el método histórico-lógico y el dialéctico para el estudio crítico de los trabajos anteriores y para utilizar éstos como punto de referencia y comparación de los resultados alcanzados.

Métodos lógicos: El método analítico-sintético, al descomponer la investigación en elementos por separado y profundizar en el estudio de cada uno de ellos, para luego sintetizarlos en la solución de la propuesta; el método inducción-deducción, como vía de la constatación teórica durante el desarrollo de la investigación.

Métodos empíricos: El método coloquial para la presentación y discusión de los resultados; el método experimental para comprobar la utilidad de los resultados obtenidos y la comparación con otros métodos reportados.

Desarrollo

Detección de intrusos a partir del mal uso

Bajo el enfoque de detección de intrusos a partir del mal uso (Lunt, 1993), las intrusiones se detectan comparando el comportamiento real registrado con patrones conocidos como sospechosos. Este enfoque resulta eficaz en el descubrimiento de ataques conocidos, pero es inútil cuando se enfrentan a variantes de ataques desconocidas, es decir, variantes de ataques de los cuales no se tiene firma (Idrees *et al.,* 2013; Kim *et al.,* 2014). Cualquier error en la definición de estas firmas aumenta la tasa de falsas alarmas y disminuye la eficacia de la técnica de detección. El mismo consta de cuatro componentes: la colección de datos, el perfil del sistema, detección de uso indebido y la respuesta. Los datos se recogen de una o varias fuentes de datos, incluyendo, el tráfico de red, las trazas de llamadas al sistema, etc., esos datos recogidos se estandarizan a un formato comprensible por los demás componentes del

sistema. Por otra parte el perfil de sistema se utiliza para caracterizar los comportamientos normales y anormales (Garcia-Teodoro *et al.*, 2009).

Detección basada en anomalías

A diferencia de la detección a partir de mal uso, la detección de anomalías se dedica a establecer los perfiles de actividad normal para el sistema. Se basa en la suposición de que todas las actividades intrusivas son necesariamente anómalas. Los estudios de detección de anomalías empiezan definiendo cuales son los atributos normales de los objetos observados, para determinar cuáles son las actividades anómalas (Agrawal *et al.*, 2013; DeOrio *et al.*, 2013; Eskin *et al.*, 2013). Un modelo de detección de anomalías consta de cuatro componentes: la recopilación de datos, el perfil normal del sistema, detección de anomalías y la respuesta. Las actividades normales del usuario o de tráfico de datos se obtienen y se guardan por el componente de recolección de datos. Técnicas específicas de modelado se utilizan para crear perfiles normales del sistema. El componente de detección de anomalías determina en qué medida las actividades actuales se desvían de los perfiles normales del sistema y que porcentaje de estas actividades debe ser marcado como anormal. Finalmente, el componente de respuesta informa sobre la intrusión. La principal ventaja de la detección de anomalías es su capacidad para encontrar nuevos ataques, como tal, se refiere a la limitación más grande de la detección de mal uso. Sin embargo, debido a los supuestos que subyacen a los mecanismos de detección de anomalías, sus tasas de falsas alarmas son en general muy altas. Específicamente, las principales razones para esta limitación son:

- El modelo de comportamiento de usuario normal se basa en datos capturados durante un período de funcionamiento normal, las actividades intrusivas perdidas durante este período son susceptibles de ser consideradas como conductas normales.
- Las técnicas de detección de anomalías difícilmente pueden detectar ataques furtivos, porque este tipo de ataques se encuentra oculto en gran número de casos de comportamientos normales.

Además, los tipos de parámetros utilizados como entradas de los modelos normales son generalmente decididos por expertos en seguridad. Cualquier error que ocurra durante el proceso de definición de estos parámetros aumenta la tasa de falsas alarmas y por lo tanto disminuye la eficacia del sistema de detección de anomalías. Como resultado, el diseño de los métodos de detección y la selección de los atributos del sistema o la red a ser monitoreados son dos de las principales cuestiones abiertas en la detección de anomalías. Muchas técnicas de detección de anomalías se han propuesto en la literatura. Estos van desde modelos estadísticos avanzados para la inteligencia artificial hasta modelos

biológicos sobre la base de los sistemas inmunológicos humanos (Garcia-Teodoro, Diaz-Verdejo, Maciá-Fernández and Vázquez, 2009). Las técnicas de detección de anomalías acorde con el tipo de procesamiento para obtener el modelo de comportamiento de un sistema, pueden ser clasificadas en tres categorías principales (Lazarevic *et al.*, 2005).

- Estadísticas.
- Basadas en conocimiento.
- Aprendizaje automático.

En el caso de las técnicas estadísticas el comportamiento del sistema es representado desde un punto de vista aleatorio. Por otra parte, las técnicas basadas en conocimiento tratan de recrear el comportamiento a partir de sistemas de datos disponibles (especificaciones de protocolos, instancias de tráfico de red, etc.). Por último, las técnicas de aprendizaje automático están basadas en un modelo implícito o explícito que permite categorizar los patrones analizados (Garcia-Teodoro, Diaz-Verdejo, Maciá-Fernández and Vázquez 2009; Kaur, Singh and Minhas, 2013). En este artículo nos centramos en las técnicas de detección basadas en aprendizaje automático. En la siguiente sección se abordan los detalles.

Esquemas de NIDS basados en aprendizaje automático

Las técnicas de aprendizaje automático están basadas sobre un modelo explícito o implícito establecido que posibilita categorizar los patrones analizados. Una característica singular de estos esquemas es la necesidad de datos etiquetados para entrenar el modelo de comportamiento, siendo este un procedimiento que demanda recursos. Muchos esquemas basados en aprendizaje automático han sido aplicados a NIDS. Algunos de los más importantes son Redes Bayesianas, Modelos de Markov, Redes Neuronales, Técnicas de lógica difusa, Algoritmos genéticos, Agrupamiento y detección de outlier (Garcia-Teodoro, Diaz-Verdejo, Maciá-Fernández and Vázquez, 2009; Kaur, Singh and Minhas, 2013; Patel, Patel and Patel, 2013).

Además de esas técnicas, existen otras que ayudan en la tarea de tratar con los grandes volúmenes de información contenidos en los juegos de datos, conocidas como técnicas de reducción de dimensionalidad (Vishwakarma *et al.*, 2013). Dos de esas técnicas son: Análisis de Componentes Principales (PCA) (Zhao *et al.*, 2013) que se basa en la reducción de la dimensionalidad a partir de transformaciones aplicadas a los datos y la selección de atributos (Ahmed 2014; Ma *et al.*, 2014; Song *et al.*, 2013), que es la aplicación de técnicas de aprendizaje automático y de búsqueda, para seleccionar un subconjunto de atributos con el objetivo de reducir el volumen de datos y aumentar el rendimiento

de los algoritmos aplicados, ganando en velocidad y a partir del cual se obtienen mejores resultados de clasificación. Los NIDS basados en técnicas de aprendizaje automático tienen la siguiente taxonomía:

Taxonomía de los sistemas de detección de anomalías

La idea de aplicar técnicas de aprendizaje automático para la detección de intrusos consiste en la construcción automática de modelos basados en el conjunto de datos de entrenamiento (Kaur, Singh and Minhas, 2013). Este conjunto de datos contiene una colección de instancias de datos los cuales pueden ser descritos mediante un conjunto de atributos (características) y las etiquetas de clasificación asociadas. Los atributos pueden ser de diferentes tipos, tales como nominales o continuos. La naturaleza de atributos determina la aplicabilidad de las técnicas de detección de anomalías. Por ejemplo, los métodos basados en distancia son inicialmente construidos para trabajar con atributos continuos y por lo general no ofrecen resultados satisfactorios con atributos nominales. Las etiquetas asociadas a las instancias de datos son generalmente en forma de valores binarios, es decir, normal (no ataque) y anómala (ataque). Por el contrario, algunos investigadores han empleado diferentes tipos de ataques, como DoS, U2R, R2L, Probe en lugar de la etiqueta anómala. De esta manera las técnicas de aprendizaje son capaces de proporcionar más información acerca de los tipos de las anomalías. Sin embargo, los resultados experimentales muestran que las técnicas actuales de aprendizaje no son suficientemente precisas como para reconocer los tipos de anomalías. Dado que el etiquetado se hace a menudo manualmente por expertos humanos, la obtención de un correcto conjunto de datos etiquetados que sea representativo de todos los tipos de comportamientos es bastante difícil (Garcia-Teodoro, Diaz-Verdejo, Maciá-Fernández and Vázquez, 2009). Las técnicas basadas en aprendizaje automático aprenden a partir de atributos determinados por el tráfico de red, de ahí la importancia de conocer los mismos.

Atributos de tráfico de red

Una de las fases más importantes en el diseño de sistemas de detección de intrusos es la identificación del conjunto de atributos a utilizar. La selección influye directamente en el rendimiento del sistema y en los tipos de ataques que el mismo detectará. Pero existe confusión general entorno a cuáles son los mejores atributos de red, debido a muchas causas; una de ellas es la carencia de un esquema de clasificación universalmente aceptado.

Esquema de clasificación para atributos de conexiones

En (Onut and Ghorbani, 2007) presentan un esquema de clasificación de atributos para detección de intrusos en redes. De esta forma se logra un mejor entendimiento sobre los atributos que pueden ser extraídos de los paquetes de red.

Bajo este esquema se logran agrupar los atributos para detectar tipos de ataques específicos. La mayoría de los artículos científicos hacen una distinción entre atributos obtenidos respecto a una conexión TCP simple y los atributos que son obtenidos considerando múltiples conexiones TCP:

- Atributos TCP Básicos: son aquellos atributos que caracterizan una conexión TCP/IP simple. Los nombres para esta categoría difieren entre los autores. En (Dokas *et al.*, 2002; Ertoz *et al.*, 2003) usan el nombre de Atributos Básicos; (Lee *et al.*, 1999) usan Atributos Esenciales; KDD Cup 99 usa Atributos Básicos de una conexión TCP individual, mientras (Lichodzijewski *et al.*, 2002) propone Atributos Básicos TCP como nombre para esta categoría. Finalmente (Mahoney and Chan, 2003) usa el nombre Flujos Estadísticos para un super conjunto de esta categoría, la cual incluye protocolos no orientados a la conexión como UDP, ICMP.
- Atributos Derivados: Son aquellos atributos que caracterizan múltiples conexiones TCP/IP al mismo tiempo (Dokas, Ertoz, Kumar, Lazarevic, Srivastava and Tan, 2002; Ertoz, Eilertson, Lazarevic, Tan, Dokas, Kumar and Srivastava, 2003). En (Lee, Stolfo and Mok, 1999) son conocidos como Atributos de Tráfico. Por su uso los sistemas encuentran similitudes que existen en la red entre diferentes conexiones TCP. Para obtener los Atributos Derivados se usan dos tipos de ventanas deslizantes. Un primer enfoque usa una ventana de tiempo con un intervalo de unos pocos segundos (Ej. 5 segundos), mientras que el segundo enfoque usa una ventana de conexión con un intervalo de varias conexiones (Ej. 100 conexiones). Esta categoría se divide a su vez en: Atributos basados en Tiempo: Incluyen todos los atributos derivados obtenidos con respecto a los x segundos pasados (donde x es el tamaño de la ventana de tiempo). Atributos basados en Conexión: Incluyen todos los atributos derivados obtenidos con respecto a las últimas k conexiones TCP encontradas en la red. (Dokas, Ertoz, Kumar, Lazarevic, Srivastava and Tan, 2002; Ertoz, Eilertson, Lazarevic, Tan, Dokas, Kumar and Srivastava, 2003; Lee, Stolfo and Mok, 1999)

Mientras que la primera categoría de atributos (Atributos TCP Básicos) son usados para caracterizar y detectar ataques que usan una única conexión, la segunda categoría es usada para detectar ataques que emplean múltiples conexiones al mismo tiempo (Ej. Scanning, DDoS, Gusanos). Específicamente los atributos dentro de la categoría de Atributos basados en Tiempo son utilizados en la detección de ataques que ocurren en un corto intervalo de tiempo tales como gusanos y DDoS. Por último Atributos basados en Conexión son usados para la detección de ataques que ocurren en un largo período de tiempo, usualmente varios minutos o incluso horas.

Preprocesamiento de datos

El preprocesamiento de datos es requerido en todas las tareas de descubrimiento de conocimiento, incluyendo tareas de detección de intrusos en redes, el cual intenta clasificar el tráfico de red como normal o anómalo (Ahmed 2014; Ihsan *et al.*, 2013; Ma, Liao and Yuan 2014; Rouhi *et al.*, 2013; Song, Zhu, Scully and Price 2013; Vishwakarma, Jain and Jain, 2013; Zhao, Kang and Kim, 2013). Varios modelos de procesos formales han sido propuestos para el descubrimiento de conocimiento y minería de datos (KDDM), tal como fue revisado por (Kurgan and Musilek, 2006). Estos modelos estiman que la etapa de preprocesamiento de datos toma el 50% del esfuerzo total del proceso, mientras que la tarea de minería de datos tarda entre el 10% y el 20%. Los pasos de preprocesamiento de datos incluyen la creación de conjunto de datos, limpieza de datos, integración, construcción de atributos para derivar nuevas funciones de nivel superior, selección de atributos para seleccionar el subconjunto óptimo de atributos relevantes, reducción, y discretización (Kotsiantis *et al.*, 2006).

Los pasos más relevantes para NIDS son:

- Creación del conjunto de datos: implica identificar el tráfico de red a usar para entrenamiento y para prueba. Estos conjuntos de datos deben ser etiquetados indicando si la conexión es normal o es anómala. La tarea de etiquetar las instancias de tráfico de red tienen asociado un alto costo computacional y de tiempo, siendo esto una tarea muy complicada.
- Construcción de atributo: tiene como propósito crear atributos adicionales con una mejor capacidad discriminativa que el conjunto inicial de atributos. Esto puede significar mejoras en los resultados de los algoritmos de aprendizaje automático que se aplican. Los atributos pueden ser construidos manualmente, o usando métodos de minería de datos como: análisis de secuencia, minería de asociación.
- Reducción: es comúnmente usada para disminuir la dimensión del conjunto de datos desechando cualquier atributo redundante o irrelevante. Este proceso de automatización es llamado selección de atributos y es usado para aliviar la dimensionalidad. La reducción de datos puede ser alcanzada con la extracción de atributos transformando el conjunto inicial en un reducido número de nuevos atributos. El Análisis de Componentes Principales (PCA) es un método común usado para la reducción de datos.

El preprocesamiento convierte el tráfico de red en series de observaciones, donde cada observación es representada como un vector de atributos. Las observaciones son opcionalmente etiquetadas según su clase, por ejemplo “normal” o “anómala”. Entonces estos vectores de atributos son adecuados como entradas para algoritmos de aprendizaje automático. En (Chandola *et al.*, 2009) se centran en una revisión de los algoritmos usados por los métodos de

detección de anomalías. Discuten muchas aplicaciones de la detección de anomalías como en fraudes de tarjetas de crédito, procesamiento de imágenes, sensores de redes, así como seguridad en computadoras. En (Garcia-Teodoro, Diaz-Verdejo, Maciá-Fernández and Vázquez, 2009) listan las técnicas de detección de anomalías usadas por los NIDS disponibles tanto comerciales como proyectos de investigación. Los autores notan una tendencia sobre los proyectos de investigación de más de una década desde los primeros enfoques estadísticos, sistemas basados en conocimiento y más recientemente técnicas de aprendizaje automático con un particular uso de N-grams y modelos de Markov. En (Gogoi *et al.*, 2010) comparan algoritmos supervisados y semi-supervisados de detección de anomalías, y prueban algunas implementaciones en el juego de datos KDD Cup 99. Ninguna de esas revisiones ha listado o comparado las técnicas de preprocesamiento de datos usadas en los NIDS basados en la detección de anomalías, refiriéndose a que atributos de red fueron escogidos como base para la detección. El preprocesamiento de los datos requiere un esfuerzo significante e impacta directamente sobre la exactitud y capacidad de los algoritmos aplicados (Kotsiantis, Kanellopoulos and Pintelas 2006; Lee and Stolfo 2000). Por lo tanto el preprocesamiento de los datos constituye una cuestión importante de los NIDS basados en la detección de anomalías. Esto se ve aún más motivado por el hecho de que ataques basados en contenidos están siendo más relevantes, mientras que antiguos ataques como DoS, escaneo o sondeo de redes están siendo atenuados. Un nuevo conjunto de técnicas de preprocesamiento es requerido para detectar esos ataques basados en contenido.

En (Chowdhary *et al.*, 2014; Davis and Clark, 2011) realizan una revisión de los NIDS atendiendo a sus datos de entrada, técnicas de preprocesamiento, algoritmos utilizados y ataques detectados. Entre ellos se destacan:

- NIDS que analizan atributos básicos de la cabecera de los paquetes: PHAD, SPADE (Guennoun *et al.*, 2008; Staniford *et al.*, 2002).
- NIDS que analizan atributos de cabecera derivados de una conexión simple (Estevez-Tapiador *et al.*, 2003; Ramadas *et al.*, 2003; Yamada *et al.*, 2007; Zhao *et al.*, 2009).
- NIDS que analizan atributos de cabecera derivados de múltiples conexiones (Lu and Ghorbani 2009; Patcha and Park, 2007; Wang and Stolfo, 2004).
- NIDS que analizan el contenido de los datos de los paquetes dirigidos a los servidores: PAYL (Wang and Stolfo 2004), POSEIDON (Bolzoni *et al.*, 2005), McPAD (Kiani *et al.*, 2008; Kloft *et al.*, 2008; Perdisci *et al.*, 2009; Rieck and Laskov, 2007).
- NIDS que analizan el contenido de los datos de los paquetes de ataques dirigidos a los clientes: JSAND, Monkey, Noxes (Cova *et al.*, 2010; Chen *et al.*, 2009; Feinstein *et al.*, 2007; Kirda *et al.*, 2006).

Conjuntos de Datos. KDD Cup 99

En (Shiravi *et al.*, 2012) proponen el ISCX 2012 Intrusion Detection Evaluation Dataset, comparándolo con los demás juegos de datos existentes, teniendo en cuenta una serie de características. El mismo está formado por 19 atributos, incluyendo atributos de contenido. Hasta el momento no se encuentra ningún artículo que reporte el uso de este juego de datos en tareas de detección de anomalías. Muchos artículos hacen uso de KDD Cup 99 como datos etiquetados para probar y comparar algoritmos de detección de intrusos (Siddiqui and Naahid, 2013). En (Mahoney and Chan, 2003) caracterizan sus desventajas. Pero es un conjunto de datos públicamente disponible, etiquetado y preprocesado, para los algoritmos de aprendizaje automático. Esto abre el campo a los investigadores que desean probar sus algoritmos y hacer valiosas comparaciones con otros algoritmos de detección de intrusos. Generar etiquetas precisas para los conjuntos de datos es un proceso que consume mucho tiempo por tal motivo se utiliza este conjunto de datos a pesar de los años de creado que tiene. El conjunto de datos fue generado a partir de DARPA 98. Cada conexión de red fue procesada en vectores etiquetados de 41 atributos. Estos fueron construidos usando técnicas de minería de datos y sistemas expertos. Los datos preprocesados produjeron:

- 9 atributos de cabecera básicos y derivados de una conexión simple, para cada conexión.
- 9 atributos de cabecera basados en tiempo, derivados de múltiples conexiones, construidos sobre una ventana deslizante de 2 segundos.
- 10 atributos de cabecera basados en host[1], derivados de múltiples conexiones, construidos sobre una ventana deslizante de 100 conexiones que permite detectar ataques de escaneo.
- 13 atributos basados en contenido construidos a partir del contenido (payload) de los paquetes. Fueron diseñados para detectar específicamente ataques del tipo U2R y R2L.

Como se puede apreciar en la Tabla 1, El conjunto de datos KDD Cup 99 contiene alrededor de 5 millones de instancias, donde cada una representa una conexión TCP/IP que está compuesta por 41 atributos tanto cuantitativos como cualitativos. En muchas investigaciones se utiliza una pequeña porción que representa el 10 % del juego de datos original, contiene 494021 instancias. Este subconjunto es utilizado para entrenamiento, mientras que para prueba se utiliza otro subconjunto que contiene 331029 instancias. Aproximadamente el 20% de ambos subconjuntos representan patrones normales de tráfico (no ataques). El juego de datos en su totalidad contiene 39 tipos de ataques agrupados en 4 categorías.

[1] El término ***host*** ("**anfitrión**", en español) es usado en informática para referirse a las computadoras conectadas a una red, que proveen y utilizan servicios de ella.

Revista Cubana de Ciencias Informáticas
ISSN: 2227-1899 | RNPS: 2301
Vol. 8, No. 4, Octubre-Diciembre, 2014
Págs. 52-73
http://rcci.uci.cu

El problema de clasificación en el conjunto KDD Cup 99 puede ser tratado bajo dos enfoques:

- Binario: consiste en distinguir entre ataque y no ataque.
- Clases múltiples clases: consiste en distinguir los tipos de ataques.

El juego de datos KDD Cup 99 es ampliamente utilizado bajo un enfoque de flujos de datos para evaluar varios algoritmos de clasificación.

Tabla 1 Características básicas de KDD Cup 99. (Bolón-Canedo *et al.*, 2011)

Dataset	DoS	Probe	U2R	R2L	Normal
10 % KDD	391458	4107	52	1126	97277
Corrected KDD	229853	4166	70	16347	60593
Whole KDD	3883370	41102	52	1126	972780

Como se muestra en la Tabla 2 el porciento de ataques en ambos conjuntos de datos es elevado. La mayoría de los ataques pertenecen a la categoría DoS. A pesar de esto, el conjunto de datos está muy desbalanceado respecto a determinadas categorías como U2R y R2L, de las cuales se contienen muy pocos ejemplos. Otras investigaciones han manipulado estas deficiencias y han creado un conjunto de datos basado en KDD Cup 99 llamado NSL-KDD, del cual eliminaron instancias duplicadas (78% y 75% de instancias duplicadas en los conjuntos de entrenamiento y prueba respectivamente). Esto provoca que el conjunto de datos pierda su sentido real, debido a que en entornos reales se repiten instancias, y los métodos implementados deben tener esto en cuenta.

Tabla 2 Distribución de tráfico normal y ataques en el KDD Cup 99. (Bolón-Canedo, Sánchez-Maroño and Alonso-Betanzos 2011)

Tipo	% conjunto de entrenamiento	% conjunto de prueba
Normal	19.69	19.48
DoS	79.24	73.90
Probe	0.83	1.34
R2L	0.23	5.21
U2R	0.01	0.07

El subconjunto utilizado para entrenar es un buen candidato para la selección de atributos debido a sus características (ver Tabla 3):

- Dos atributos constantes (num_outbound_cmds e is_host_login).

- Otros atributos casi constantes (land, root_shell, num_shells).
- Atributos continuos desbalanceados los cuales pudieran ser discretizados.

Tabla 3 Atributos continuos desbalanceados en KDD Cup 99. (Bolón-Canedo, Sánchez-Maroño and Alonso-Betanzos 2011)

Atributo	Valor mínimo	Valor máximo	Media	Desviación estándar
duration	0	58,329	47.98	707.75
src_byes	0	693,375,640	3025.61	3025.61
dst_bytes	0	5,155,468	868.53	33040.00

Selección de atributos en KDD Cup 99

La selección de atributos consiste en determinar los atributos relevantes y desechar los irrelevantes, con el objetivo de obtener un subconjunto de atributos que describa correctamente el problema o proceso en cuestión sin afectar el rendimiento de los algoritmos. La selección tiene ventajas como: (Guyon and Elisseeff, 2003)

- Mejora el rendimiento de los algoritmos de aprendizaje automático.
- Reducción de dimensionalidad.
- Posibilita usar modelos simples ganando así en velocidad.

Existen dos enfoques para la selección de atributos: (Kohavi and John, 1997)

- Métodos basados en filtro.
- Métodos basados en envoltorio.

Mientras que los métodos de envoltorio optimizan determinado algoritmo como parte del proceso de selección, los métodos de filtros se basan en la característica general de entrenar para seleccionar atributos con independencia del algoritmo de clasificación.

Sin embargo, los modelos de envoltorio consumen mucho tiempo, lo cual restringe su uso en grandes conjuntos de datos. Por otra parte, los métodos de filtro son menos costosos computacionalmente y tienen la posibilidad de ser aplicados a grandes conjuntos de datos. Además pueden ser más generalizados debido a que actúan independientemente del algoritmo de inducción.

Entre los métodos de filtro aplicados a KDD Cup 99 se encuentran (Bolón-Canedo, Sánchez-Maroño and Alonso-Betanzos, 2011) (ver Tabla 4):

- Correlation-based Feature Selection (CFS).
- INTERACT.
- Consistency-based.

CFS es uno de los filtros mejor conocidos y más utilizados. INTERACT es un nuevo enfoque basado en la interacción entre atributos y Consistency-based es uno de los algoritmos clásicos.

En (Mukherjee and Sharma, 2012) evalúan tres métodos para selección de atributos:

- Correlation-based Feature Selection.
- Information Gain and Gain Ratio.
- Feature Vitality Based Reduction Method.

Tabla 4 Atributos de KDD Cup 99 seleccionados por cada método. (Bolón-Canedo, Sánchez-Maroño and Alonso-Betanzos, 2011)

Algoritmo de selección	Cantidad de atributos seleccionados	Atributos seleccionados
CFS+ BestFirst	10	3, 4, 5, 6, 12, 26, 29, 30, 37, 38
GR + Ranker	14	3, 4, 5, 6, 11, 12, 22, 25, 26, 29, 30, 37, 38, 39
InfoGain + Ranker	20	3, 4, 5, 6, 12, 23, 24, 25, 26, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39
FVBRM	24	1, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 23, 24, 32, 33, 36, 38, 40

Algoritmos de discretización aplicados a KDD Cup 99

Muchos algoritmos de filtro trabajan sobre datos discretos. Por tal motivo una práctica común para estos algoritmos es discretizar los datos antes de realizar la selección. Los datos discretos son más fáciles de entender, usar, explicar, y la discretización puede el aprendizaje más preciso y rápido (Liu *et al.*, 2002). Un conjunto de algoritmos sólo trabajan con datos discretos. En la literatura aparecen varios métodos para discretizar (Ihsan, Idris and Abdullah, 2013), por ejemplo Weka (Witten and Frank, 2005) discretiza los datos empleando Entropy Minimization Discretization (Dougherty *et al.*, 1995). Atendiendo a que KDD Cup 99 es considerado un conjunto de datos de alta dimensionalidad se han aplicado algoritmos de discretización adecuados y clásicos como: (Bolón-Canedo, Sánchez-Maroño and Alonso-Betanzos, 2011)

- EMD (Entropy Minimization Discretization)
- EWD (Equal Width Discretization)
- EFD (Equal Frequency Discretization)
- PKID (Es un nuevo enfoque muy adecuado para grandes conjuntos de datos)

En (Davis and Clark, 2011) realizan una revisión de los NIDS que utilizan atributos del juego de datos KDD Cup 99, atendiendo a las técnicas de preprocesamiento, algoritmos utilizados y ataques detectados. Entre ellos se destacan: (Chebrolu *et al.,* 2005; Hernández-Pereira *et al.,* 2009; Kuang 2007; Laskov *et al.,* 2005; Li *et al.,* 2009; Wang and Battiti 2006; Yeung and Chow, 2002; Zhang and Zulkernine, 2006).

Técnicas de inteligencia de enjambre aplicadas a tareas de detección de intrusos en redes.

Para mejorar las tareas durante la detección de intrusos y el rendimiento del NIDS, han sido realizadas investigaciones que utilizan técnicas de inteligencia de enjambre tanto para optimizar la detección como para mejorar las respuestas del sistema. Varios artículos (Kolias *et al.,* 2011; Satpute *et al.,* 2013) han demostrado que hibridar algoritmos de aprendizaje automático con algoritmos de inteligencia de enjambre mejora la detección de anomalías con respecto a otros enfoques y han propuesto:

- ACO (Ant Colony Optimization) orientado a IDS (ACO para detectar el origen del ataque, ACO para la inducción de reglas de clasificación).
- PSO (Particle Swarm Optimization) orientado a IDS (PSO & Redes Neuronales, PSO & SVM (Support Vector Machine), PSO & K-Means, PSO para la inducción de reglas de clasificación).
- ACC (Ant Colony Clustering) orientado a IDS (ACC & SOM (Self-organizing map), ACC & SVM).

La mayoría de los enfoques que utilizan ACO, lo hacen como mecanismo de respuestas, por ejemplo para determinar desde donde es la intrusión. Es menos común su utilización en la etapa de detección. Por otra parte PSO no es utilizado como un mecanismo de clasificación puro, la tendencia es hibridarlo con algoritmos de clasificación y ha demostrado mejorar el rendimiento de todas las técnicas de aprendizaje automático con las que ha sido probado. ACC ha dado los mejores resultados de clasificación para la mayoría de las clases de ataques, sobre todo para ataques del tipo R2L. Sugieren que sería interesante el estudio de la hibridación entre ACC con algoritmos de clasificación (Kolias, Kambourakis and Maragoudakis, 2011).

Discusión

La aplicación de métodos generales como el método hipotético-deductivo permitió establecer estrategias de investigación y definir líneas de trabajo sobre la detección de intrusos en redes de computadoras. Esto, unido a los métodos dialéctico e histórico-lógico permitió hacer el estudio de trabajos anteriores, en busca de tendencias, utilizándolos como puntos de referencias para la determinación de posibles áreas de aporte investigativo en la

aplicación de técnicas de aprendizaje automático para la detección de intrusos en las redes. A su vez, métodos lógicos como el análisis y la síntesis facilitaron descomponer la detección de intrusos en detección basada en firmas y basada en anomalías, profundizando en la detección basada en anomalías y dividiéndola en enfoques basados en aprendizaje automático, en la taxonomía de los sistemas, así como en las técnicas de preprocesamiento y procesamiento empleadas sobre los datos; sintetizando los resultados obtenidos. Específicamente en la etapa de preprocesamiento se aplicaron métodos empíricos como el experimental para comprobar los resultados de los algoritmos de selección y discretización de atributos propuestos en otras investigaciones. Se aplicó el método coloquial para la presentación de los resultados en las etapas de preprocesamiento de los datos para la detección de anomalías. Utilizando ese método se presentaron los resultados de varios algoritmos de selección y discretización aplicados sobre el conjunto de datos KDD Cup 99. Luego, en la etapa de procesamiento, la cual constituye la detección de anomalías en sí, se logran mejores resultados en cuanto a precisión y tasas de falsos positivos a partir de hibridar algoritmos de aprendizaje automático con algoritmos de optimización basados en inteligencia de enjambre.

Conclusiones

Con el estudio realizado se logró determinar que el diseño de los métodos de detección y la selección de los atributos del sistema o la red a ser monitoreados son dos de las principales cuestiones abiertas en la detección de anomalías. Además de eso se identifican necesidades que dan lugar a áreas de trabajo como la definición de nuevos atributos de red basados en contenido para construir modelos de aprendizaje que permitan detectar los nuevos ataques basados en contenido. La realización de nuevos experimentos que hibriden algoritmos de inteligencia de enjambre con algoritmos de aprendizaje automático, para mejorar los índices de detección en ataques basados en contenido como las variantes de U2R y R2L son cuestiones en las que se trabajan actualmente, fundamentalmente en experimentos que hibridan Ant Colony Optimization, debido a que es una metaheurística menos costosa computacionalmente, con algoritmos de aprendizaje automático.

Referencias

– AGRAWAL, H., C. BEHRENS AND B. DASARATHY. Learning program behavior for anomaly detection. In.: Google Patents, 2013.
– AHMED, P. A Hybrid-Based Feature Selection Approach for IDS. In *Networks and Communications (NetCom2013).* Springer, 2014, p. 195-211.

- BOLÓN-CANEDO, V., N. SÁNCHEZ-MAROÑO AND A. ALONSO-BETANZOS Feature selection and classification in multiple class datasets: An application to KDD Cup 99 dataset. Expert Systems with Applications, 2011, 38(5), 5947-5957.
- BOLZONI, D., E. ZAMBON, S. ETALLE AND P. HARTEL Poseidon: A 2-tier anomaly-based intrusion detection system. arXiv preprint cs/0511043, 2005.
- CISCO. Get the Latest Findings on Malware Threats. In., 2014.
- COVA, M., C. KRUEGEL AND G. VIGNA. Detection and analysis of drive-by-download attacks and malicious JavaScript code. In *Proceedings of the 19th international conference on World wide web.* ACM, 2010, p. 281-290.
- CHANDOLA, V., A. BANERJEE AND V. KUMAR Anomaly detection: A survey. ACM Computing Surveys (CSUR), 2009, 41(3), 15.
- CHEBROLU, S., A. ABRAHAM AND J. P. THOMAS Feature deduction and ensemble design of intrusion detection systems. Computers & Security, 2005, 24(4), 295-307.
- CHEN, C.-M., W.-Y. TSAI AND H.-C. LIN. Anomaly behavior analysis for web page inspection. In *Networks and Communications, 2009. NETCOM'09. First International Conference on.* IEEE, 2009, p. 358-363.
- CHOWDHARY, M., S. SURI AND M. BHUTANI Comparative Study of Intrusion Detection System 2014.
- DAVIS, J. J. AND A. J. CLARK Data preprocessing for anomaly based network intrusion detection: A review. Computers & Security, 2011, 30(6), 353-375.
- DEORIO, A., Q. LI, M. BURGESS AND V. BERTACCO. Machine learning-based anomaly detection for post-silicon bug diagnosis. In *Proceedings of the Conference on Design, Automation and Test in Europe.* EDA Consortium, 2013, p. 491-496.
- DOKAS, P., L. ERTOZ, V. KUMAR, A. LAZAREVIC, *et al.,* Data mining for network intrusion detection. In *Proc. NSF Workshop on Next Generation Data Mining.* 2002, p. 21-30.
- DONG, G., J. GAO, R. DU, L. TIAN, *et al.,* Robustness of network of networks under targeted attack. Physical Review E, 2013, 87(5), 052804.
- DOUGHERTY, J., R. KOHAVI AND M. SAHAMI. Supervised and unsupervised discretization of continuous features. In *ICML.* 1995, p. 194-202.
- ERTOZ, L., E. EILERTSON, A. LAZAREVIC, P.-N. TAN, *et al.,* Detection of novel network attacks using data mining. In *Proc. of Workshop on Data Mining for Computer Security.* Citeseer, 2003.

- ESKIN, E., A. O. ARNOLD, M. PRERAU, L. PORTNOY, *et al.*, Methods of unsupervised anomaly detection using a geometric framework. In.: Google Patents, 2013.
- ESTEVEZ-TAPIADOR, J. M., P. GARCIA-TEODORO AND J. E. DIAZ-VERDEJO. Stochastic protocol modeling for anomaly based network intrusion detection. In *Information Assurance, 2003. IWIAS 2003. Proceedings. First IEEE International Workshop on.* IEEE, 2003, p. 3-12.
- FEINSTEIN, B., D. PECK AND I. SECUREWORKS Caffeine monkey: Automated collection, detection and analysis of malicious javascript. Black Hat USA, 2007, 2007.
- GARCIA-TEODORO, P., J. DIAZ-VERDEJO, G. MACIÁ-FERNÁNDEZ AND E. VÁZQUEZ Anomaly-based network intrusion detection: Techniques, systems and challenges. Computers & Security, 2009, 28(1), 18-28.
- GOGOI, P., B. BORAH AND D. BHATTACHARYYA Anomaly detection analysis of intrusion data using supervised & unsupervised approach. Journal of Convergence Information Technology, 2010, 5(1), 95-110.
- GUENNOUN, M., A. LBEKKOURI AND K. EL-KHATIB. Selecting the best set of features for efficient intrusion detection in 802.11 networks. In *Information and Communication Technologies: From Theory to Applications, 2008. ICTTA 2008. 3rd International Conference on.* IEEE, 2008, p. 1-4.
- GUYON, I. AND A. ELISSEEFF An introduction to variable and feature selection. The Journal of Machine Learning Research, 2003, 3, 1157-1182.
- HEADY, R., G. LUGER, A. MACCABE AND M. SERVILLA *The architecture of a network-level intrusion detection system*. Edtion ed.: Department of Computer Science, College of Engineering, University of New Mexico, 1990.
- HERNÁNDEZ-PEREIRA, E., J. A. SUÁREZ-ROMERO, O. FONTENLA-ROMERO AND A. ALONSO-BETANZOS Conversion methods for symbolic features: A comparison applied to an intrusion detection problem. Expert Systems with Applications, 2009, 36(7), 10612-10617.
- IDREES, F., M. RAJARAJAN AND A. MEMON. Framework for distributed and self-healing hybrid intrusion detection and prevention system. In *ICT Convergence (ICTC), 2013 International Conference on.* IEEE, 2013, p. 277-282.
- IHSAN, Z., M. Y. IDRIS AND A. H. ABDULLAH Attribute Normalization Techniques and Performance of Intrusion Classifiers: A Comparative Analysis. Life Science Journal, 2013, 10(4).
- KAUR, H., G. SINGH AND J. MINHAS A Review of Machine Learning based Anomaly Detection Techniques. arXiv preprint arXiv:1307.7286, 2013.

- KIANI, M., A. CLARK AND G. MOHAY. Evaluation of anomaly based character distribution models in the detection of SQL injection attacks. In *Availability, Reliability and Security, 2008. ARES 08. Third International Conference on.* IEEE, 2008, p. 47-55.
- KIM, G., S. LEE AND S. KIM A novel hybrid intrusion detection method integrating anomaly detection with misuse detection. Expert Systems with Applications, 2014, 41(4), 1690-1700.
- KIRDA, E., C. KRUEGEL, G. VIGNA AND N. JOVANOVIC. Noxes: a client-side solution for mitigating cross-site scripting attacks. In *Proceedings of the 2006 ACM symposium on Applied computing.* ACM, 2006, p. 330-337.
- KLOFT, M., U. BREFELD, P. DÜESSEL, C. GEHL, *et al.*, Automatic feature selection for anomaly detection. In *Proceedings of the 1st ACM workshop on Workshop on AISec.* ACM, 2008, p. 71-76.
- KOHAVI, R. AND G. H. JOHN Wrappers for feature subset selection. Artificial intelligence, 1997, 97(1), 273-324.
- KOLIAS, C., G. KAMBOURAKIS AND M. MARAGOUDAKIS Swarm intelligence in intrusion detection: A survey. Computers & Security, 2011, 30(8), 625-642.
- KOTSIANTIS, S., D. KANELLOPOULOS AND P. PINTELAS Data Preprocessing for Supervised Leaning. Enformatika, 2006, 12.
- KUANG, L. Dnids: A dependable network intrusion detection system using the csi-knn algorithm 2007.
- KURGAN, L. A. AND P. MUSILEK A survey of Knowledge Discovery and Data Mining process models. The Knowledge Engineering Review, 2006, 21(01), 1-24.
- LASKOV, P., P. DÜSSEL, C. SCHÄFER AND K. RIECK. Learning intrusion detection: supervised or unsupervised? In *Image Analysis and Processing–ICIAP 2005.* Springer, 2005, p. 50-57.
- LAZAREVIC, A., V. KUMAR AND J. SRIVASTAVA. Intrusion detection: A survey. In *Managing Cyber Threats.* Springer, 2005, p. 19-78.
- LEE, W. AND S. J. STOLFO A framework for constructing features and models for intrusion detection systems. ACM transactions on Information and system security (TiSSEC), 2000, 3(4), 227-261.
- LEE, W., S. J. STOLFO AND K. W. MOK. Mining in a data-flow environment: Experience in network intrusion detection. In *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining.* ACM, 1999, p. 114-124.
- LI, Y., J.-L. WANG, Z.-H. TIAN, T.-B. LU, *et al.*, Building lightweight intrusion detection system using wrapper-based feature selection mechanisms. Computers & Security, 2009, 28(6), 466-475.

- LICHODZIJEWSKI, P., A. N. ZINCIR-HEYWOOD AND M. I. HEYWOOD. Dynamic intrusion detection using self-organizing maps. In *The 14th Annual Canadian Information Technology Security Symposium (CITSS)*. Citeseer, 2002.
- LIPPMANN, R., J. W. HAINES, D. J. FRIED, J. KORBA, *et al.*, The 1999 DARPA off-line intrusion detection evaluation. Computer Networks, 2000, 34(4), 579-595.
- LIU, H., F. HUSSAIN, C. L. TAN AND M. DASH Discretization: An enabling technique. Data mining and knowledge discovery, 2002, 6(4), 393-423.
- LU, W. AND A. A. GHORBANI Network anomaly detection based on wavelet analysis. EURASIP Journal on Advances in Signal Processing, 2009, 2009, 4.
- LUNT, T. F. A survey of intrusion detection techniques. Computers & Security, 1993, 12(4), 405-418.
- MA, S., H. LIAO AND Y. YUAN. Study on Rough Set Attribute Reduction in Intrusion Detection. In *Proceedings of the 9th International Symposium on Linear Drives for Industry Applications, Volume 3*. Springer, 2014, p. 507-512.
- MAHONEY, M. V. AND P. K. CHAN. An analysis of the 1999 DARPA/Lincoln Laboratory evaluation data for network anomaly detection. In *Recent Advances in Intrusion Detection*. Springer, 2003, p. 220-237.
- MUKHERJEE, S. AND N. SHARMA Intrusion detection using naive Bayes classifier with feature reduction. Procedia Technology, 2012, 4, 119-128.
- ONUT, I.-V. AND A. A. GHORBANI A Feature Classification Scheme For Network Intrusion Detection. IJ Network Security, 2007, 5(1), 1-15.
- PATCHA, A. AND J.-M. PARK An overview of anomaly detection techniques: Existing solutions and latest technological trends. Computer Networks, 2007, 51(12), 3448-3470.
- PATEL, A. M., A. PATEL AND H. R. PATEL COMPARATIVE ANALYSIS FOR MACHINE LEARNING TECHNIQUES APPLIANCE ON ANOMALY BASED INTRUSION DETECTION SYSTEM FOR WLAN. International Journal, 2013.
- PERDISCI, R., D. ARIU, P. FOGLA, G. GIACINTO, *et al.*, McPAD: A multiple classifier system for accurate payload-based anomaly detection. Computer Networks, 2009, 53(6), 864-881.
- RAMADAS, M., S. OSTERMANN AND B. TJADEN. Detecting anomalous network traffic with self-organizing maps. In *Recent Advances in Intrusion Detection*. Springer, 2003, p. 36-54.
- RIECK, K. AND P. LASKOV Language models for detection of unknown attacks in network traffic. Journal in Computer Virology, 2007, 2(4), 243-256.

- ROUHI, R., F. KEYNIA AND M. AMIRI Improving the Intrusion Detection Systems' Performance by Correlation as a Sample Selection Method. Journal of Computer Sciences and Applications, 2013, 1(3), 33-38.
- SATPUTE, K., S. AGRAWAL, J. AGRAWAL AND S. SHARMA. A Survey on Anomaly Detection in Network Intrusion Detection System Using Particle Swarm Optimization Based Machine Learning Techniques. In *Proceedings of the International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA).* Springer, 2013, p. 441-452.
- SHIRAVI, A., H. SHIRAVI, M. TAVALLAEE AND A. A. GHORBANI Toward developing a systematic approach to generate benchmark datasets for intrusion detection. Computers & Security, 2012, 31(3), 357-374.
- SIDDIQUI, M. K. AND S. NAAHID Analysis of KDD CUP 99 Dataset using Clustering based Data Mining. International Journal of Database Theory & Application, 2013, 6(5).
- SONG, J., Z. ZHU, P. SCULLY AND C. PRICE. Selecting Features for Anomaly Intrusion Detection: A Novel Method using Fuzzy C Means and Decision Tree Classification. In *Cyberspace Safety and Security.* Springer, 2013, p. 299-307.
- STANIFORD, S., J. A. HOAGLAND AND J. M. MCALERNEY Practical automated detection of stealthy portscans. Journal of Computer Security, 2002, 10(1), 105-136.
- SUNG, A. H. AND S. MUKKAMALA. Identifying important features for intrusion detection using support vector machines and neural networks. In *Applications and the Internet, 2003. Proceedings. 2003 Symposium on.* IEEE, 2003, p. 209-216.
- VISHWAKARMA, U., A. JAIN AND A. JAIN A Review of Feature Reduction in Intrusion Detection System Based on Artificial Immune System and Neural Network. INTERNATIONAL JOURNAL OF COMPUTERS & TECHNOLOGY, 2013, 9(3), 1127-1133.
- WANG, K. AND S. J. STOLFO. Anomalous payload-based network intrusion detection. In *Recent Advances in Intrusion Detection.* Springer, 2004, p. 203-222.
- WANG, W. AND R. BATTITI. Identifying intrusions in computer networks with principal component analysis. In *Availability, Reliability and Security, 2006. ARES 2006. The First International Conference on.* IEEE, 2006, p. 8 pp.
- WITTEN, I. H. AND E. FRANK *Data Mining: Practical machine learning tools and techniques*. Edtion ed.: Morgan Kaufmann, 2005. ISBN 008047702X.

– YAMADA, A., Y. MIYAKE, K. TAKEMORI, A. STUDER, *et al.*, Intrusion detection for encrypted web accesses. In *Advanced Information Networking and Applications Workshops, 2007, AINAW'07. 21st International Conference on.* IEEE, 2007, vol. 1, p. 569-576.
– YEUNG, D.-Y. AND C. CHOW. Parzen-window network intrusion detectors. In *Pattern Recognition, 2002. Proceedings. 16th International Conference on.* IEEE, 2002, vol. 4, p. 385-388.
– ZHANG, J. AND M. ZULKERNINE. Anomaly based network intrusion detection with unsupervised outlier detection. In *Communications, 2006. ICC'06. IEEE International Conference on.* IEEE, 2006, vol. 5, p. 2388-2393.
– ZHAO, J., H. HUANG, S. TIAN AND X. ZHAO. Applications of hmm in protocol anomaly detection. In *2009 International Joint Conference on Computational Sciences and Optimization.* 2009, vol. 2, p. 347-349.
– ZHAO, L., H.-S. KANG AND S.-R. KIM. Improved clustering for intrusion detection by principal component analysis with effective noise reduction. In *Information and Communication Technology.* Springer, 2013, p. 490-495.