Inefficiency of IDS Static Anomaly Detector in Real-World Networks (Diciembre 2018)

Camacho V. Steven A. Ing, *Magister en Ciencias de la Ingeniería Electrónica, USM.*

Valparaíso, Chile.

***Abstract*—El crecimiento de la tecnología en redes de comunicaciones y la vulnerabilidad que las acompaña, ha dado paso a varias investigaciones para mejorar la seguridad en dichas redes como lo son los IDS (Intrusion Detection System), los cuales consisten en analizar y bloquear intrusiones en la red por medio de alertas. Estos IDS identifican los accesos no autorizados por medio de bases datos las cuales contienen diferentes tipos de anomalías que ya han ocurrido en diferentes tipos de redes, sin embargo, estas bases de datos necesitan actualizarse constantemente debido a que los ataques cambian o mejoran. Al realizar un ataque en una red con estos nuevos parámetros y con un IDS que no está actualizado, este es no es capaz de detectar el tipo de ataque, causando una pérdida para una organización. Esto es debido a que el tráfico en una red es aleatorio se basa en sus actividades, servicios, las restricciones y aplicaciones disponibles. Este artículo presentará las diferencias entre la detección de ataques en diferentes escenarios de red mediante la aplicación de IDS tradicionales comparado con IDS usando redes neuronales artificiales (ANN).**

***Palabras clave*— IDS, redes neuronales, vulnerabilidad, seguridad.**

I. Introducción

Los sistemas de seguridad en las redes de comunicaciones requieren nuevos sistemas o algoritmos para detectar y evitar las amenazas de accesos no autorizados a los sistemas dentro de la red. Por definición intruso intenta encontrar una vulnerabilidad en el sistema para ingresar y explorarlo con el máximo acceso posible [1].

Los IDS analizan constantemente la red para reconocer los intentos de acceso no autorizados, con el fin de bloquear intentos de ataques o generar alertas para ser analizadas por una instancia inteligente [2]. Algunos ataques conocidos en algunas redes como las ad hoc se pueden clasificar en dos categorías, la primera incluye suplantación, rastreo de tráfico, modificación o reproducción [3] y la segunda se basa en ataques inherentes a redes ad hoc y que se producen incluso cuando los nodos están autenticados, Por ejemplo, un nodo autenticado puede anunciar falsos vecinos en sus mensajes de control [4,5,6].

Las bases de datos de los IDS contienen en su mayoría información estadística con respecto al uso de la red, es decir la cantidad requerimientos que realizan los usuarios pertenecientes a la misma, con esa información se entrenan los sistemas ya sean algoritmos genéticos o redes neuronales permitiendo a estos sistemas reconocer alguna variación o alteración inusual en la red [7].

Este paper propone un esquema IDS basado en redes neuronales, donde la Sección II se muestran los tipos de ataques realizados a las redes de datos, la Sección III se realiza una explicación breve de en qué se basa la selección de características, la Sección IV se describe el algoritmo a implementar por medio de aprendizaje automático, la Sección V las topologías y tipos de redes donde se implementara el sistema desarrollado, la Sección VI un análisis estadístico basado en verdaderos positivos verdaderos negativos, falsos positivos y falsos negativos, la Sección VII Conclusiones y trabajos futuros.

II. Tipos de ataques

En la sección anterior se mencionó unos tipos de ataques a las redes de comunicaciones, sin embargo, investigaciones realizadas muestran un comportamiento estadístico entre los siguientes tipos de ataques: R2L (remote to local) [8], U2R (user to root) [9], probes (sondeo) [10] y DoS (denial of services) [11]. Estos tipos de ataques presentan un patrón estadístico como se muestra en la Figura 1, pero como se mencionó anteriormente los sistemas no son capaces de detectar nuevas amenazas con respecto a ese tipo de ataques [12].

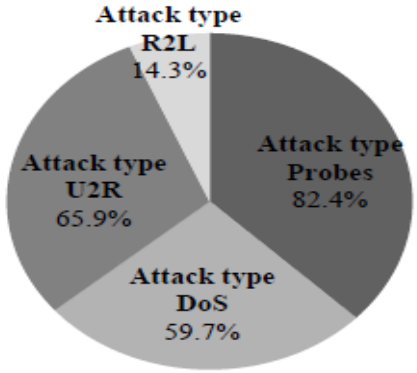

Figura 1. Porcentaje de ataques analizados con ANN [1]

Debido a esto se evaluaron las técnicas de detección de intrusos o anomalías en los sistemas en NIDS (Network Intrusion Detection System), junto con datos recolectados en DARPA (Defense Advanced Research

Projects Agency) para analizar las fallas y vulnerabilidades de los sistemas IDS y la complejidad de implementar técnicas de aprendizaje automático con datos de redes reales [12].

Con los datos obtenidos de DARPA, se ha podido concluir que los IDS pueden tener una efectividad de hasta el 74% detectando ataques de múltiples conexiones y una tasa de detección del 56% en una sola conexión [12].

Con estos resultados se han realizado análisis para diseñar IDS basado en métodos de extracción de datos [13] para identificar las ventajas y desventajas de usar sistemas híbridos con técnicas de detección de uso inadecuado y detección de anomalías [1].

III. Selección de características

La tarea principal de los algoritmos de aprendizaje automático es poder caracterizar el comportamiento normal y poderlo diferenciar de un ataque [1] a partir de un entrenamiento con modelos apropiados y de los datos proporcionados de una red ya que seleccionar buenas características es una actividad crucial y requiere un amplio conocimiento del comportamiento y variación en las redes [12].

Uno de los mayores desafíos en la detección de intrusiones basada en el aprendizaje automático es la gran cantidad de datos recopilados de la red. Por lo tanto, antes de enviar los datos a un algoritmo de aprendizaje automático, el tráfico de red sin procesar debe resumirse en eventos de nivel superior, como los registros de conexión [5]. Cada evento de nivel superior se describe con un conjunto de características.

En la TABLA I, se listan los ataques basados en diferentes conjuntos de datos [14].

TABLA I. CANTIDAD DE MUESTRAS CON RESPECTO AL TIPO DE ATAQUE Y EL CONJUNTO DE DATOS

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

Las características se seleccionaron con el programa Spleen [15], la estructura utilizada para las variables fue basada en la TABLA I [1], sin tener la misma cantidad de datos o variables que poseen.

Los datos recolectados son tomados de la red y en horas de alta congestión para analizar el uso de cada recurso en los diferentes escenarios de red; se usa el método de componentes principales [1] para extraer los atributos de los datos recolectados, esta técnica está diseñada para el análisis de bases de datos con gran cantidad de información, con esta técnica los datos son redimensionados o caracterizados por el software spleen dándoles un valor especifico a cada uno, el software separa cada una de las característica aparte de los componentes principales, se utiliza también detector de cambio de comportamiento (BCD: Behavior Change Detector), el cual tiene una ventaja para los IDS y su capacidad de detectar cambios o anomalías [16-18].

TABLA II DESCRIPCION DE CARACTERISTICAS SELECCIONADAS PARA EL ENTRENAMIENTO

Características	Descripción
Banderas TCP inusuales	Variable booleana para verificar la conexión cuando ha recibido paquetes con una configuración de marca TCP inusual.
Número de paquetes fuera de secuencia	Número de paquetes que llegan con retraso y cambian el orden del mensaje (latencia).
Promedio del tamaño de la carga útil en bytes	Este es el tamaño promedio del mensaje real sin el encabezado. El tamaño de la carga útil se muestra en una variable entera representada en la serie de bytes. Los tamaños de resultados mostrados dependen de los servicios solicitados y las conexiones activas.
Recuento de las conexiones de este cliente en la última conexión.	Esta función tiene el control de las conexiones desde el cliente-host específico solo en la última captura o análisis de datos. Esta característica es una variable entera.
Porcentaje de conexiones desde el cliente actual con los estados S0 y S1	El estado S0 es una configuración para verificar una sincronización inicial (SYN), pero después, no hay informes de una solicitud del servidor. El estado S1 es la configuración de la conexión de tres a mano, pero no muestra más paquetes entre el tráfico de red.
Porcentaje de conexiones desde el cliente actual al host actual	Esta es una característica que se muestra en una variable doble, porque representa un porcentaje de las conexiones del cliente actual que tiene el mismo host y servicios de solicitud.
Porcentaje de conexiones al host actual con el estado S0 y otros	El porcentaje de conexiones que ha intentado conectarse con el servidor, pero no ha recibido respuesta. Luego, el mismo envía otro paquete con un nuevo intento de recibir una respuesta.
Diferencia entre el número de conexiones al host actual.	Representa las conexiones rechazadas desde el host. La diferencia de host rechazada se compara con la solicitud de paquete ACK que tiene este host.
Diferencia entre el número de conexiones al servicio actual.	Representa las conexiones rechazadas, pero depende del servicio solicitado. La diferencia del servicio rechazado se compara con el servicio actual aprobado frente a los servicios solicitados.
Recuento de host sin respuesta	Esta característica se muestra en

	una variable de enteros basada en el número de conexiones no contestadas por el host. Se basa en el número de paquetes que no han llegado al destino.
Cuenta de servicio sin respuesta	Esta característica se muestra en una variable entera para el número de conexiones sin respuesta según la solicitud de servicios. Se basa en el número de paquetes que no han llegado al destino.

La Tabla II [1] tiene las caracteristicas más relevantes de un total de 42 caracteristicas que pueden encontrarse en una red, estas 11 caracteristicas tienen una mayor variacion entre sus datos para los diferentes escenarios de red, por eso fueron seleccionadas,.

Las características se muestran en líneas a través de una serie de *m* entrenamientos y *n* características. Los conjuntos de datos se generan en archivos de valores separados por comas (CSV) y contienen 52 características.

Las funciones seleccionadas son numéricas y las funciones no seleccionadas se representan como un valor nulo con el símbolo "$". Los campos nulos con las características no seleccionadas se simplifican en una nueva variable. La estructura de conexiones se representa en la Figura 2. La representación de la característica se basa en el número de conexiones y el número de características.

$$\begin{matrix} F_{11} & F_{12} & \cdots & F_{1xn} \\ F_{21} & F_{22} & \cdots & F_{2xn} \\ \vdots & \vdots & & \vdots \\ F_{mx1} & F_{mx2} & \cdots & F_{mxn} \end{matrix}$$

Figura 2 matriz con características conectadas [1]

Las variables principales son una combinación de las características originales almacenadas en una nueva matriz que representa los perfiles de red. Los escenarios de red tienen diferentes características y tráfico de datos, según los servicios frecuentes, los puertos virtuales, las solicitudes de tráfico, el número de usuarios, entre otros.

IV. TIPOS DE ALGORITMOS BASADOS EN APRENDIZAJE AUTOMÁTICO

Con los enfoques que se han realizado en detección de anomalías, se ha usado gran parte de Aprendizaje Automático para este propósito. Teniendo un gran rendimiento en la detección de anomalías, con diferentes tipos de enfoques y así poder comparar los resultados obtenidos [1].

La primera parte se realiza con la extracción de características de tráfico real en diferentes escenarios de red de es decir en redes de compañías o industrias grandes, de esta manera poder tener una correcta selección de funciones y se pueda realizar un filtrado de búsqueda. A continuación, se explica brevemente los tipos de sistemas que pueden ser implementados.

A. *ANN (Artificial Neuronal Network)*

Una de las ventajas de estos algoritmos es que al ser entrenados con diferentes entradas ellos se pueden adaptar a diferentes tipos de respuesta en las redes del mundo real [19].

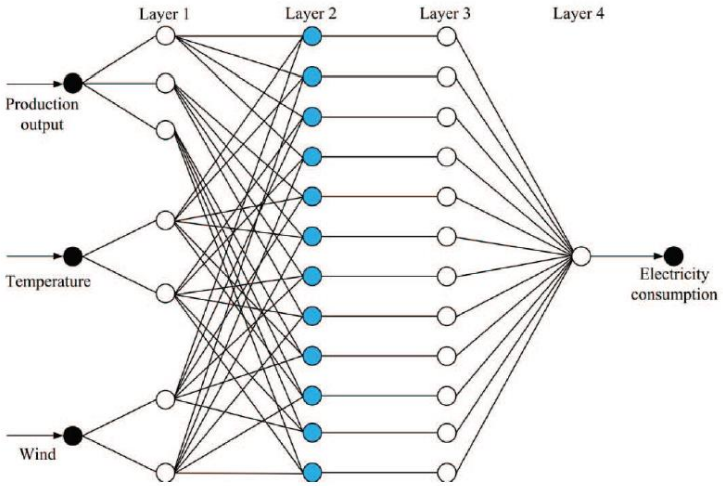

Figura 3. Topología algoritmo ANN [20].

B. *GA (Genetic Algorithms)*

Al igual que las ANN los GA son ampliamente utilizados para la detección de anomalías, estos algoritmos se basan en el estudio y cruce de características de forma que al ser combinadas se puedan tener resultados variables y tener una nueva solución en el problema provisto. Sin embargo, una desventaja es que al tener estas mutaciones pueda tener errores de detección [21].

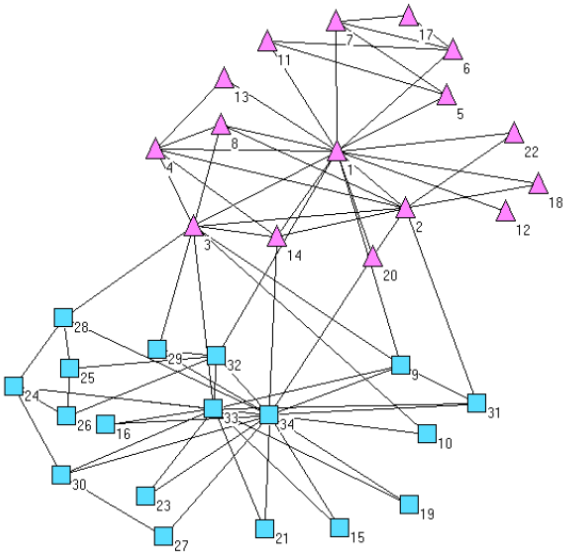

Figura 4. Diseño GA [22].

C. *SVM (Support Vector Machine)*

Este tipo de algoritmos es supervisado por lo que no se adapta a anomalías con patrones diferentes con los que ha sido entrenado provocando fallos en el mismo, sin embargo, se puede combinar con ANN o AG para el análisis de características mejorando los resultados en el análisis de anomalías o detección de intrusos [23,24].

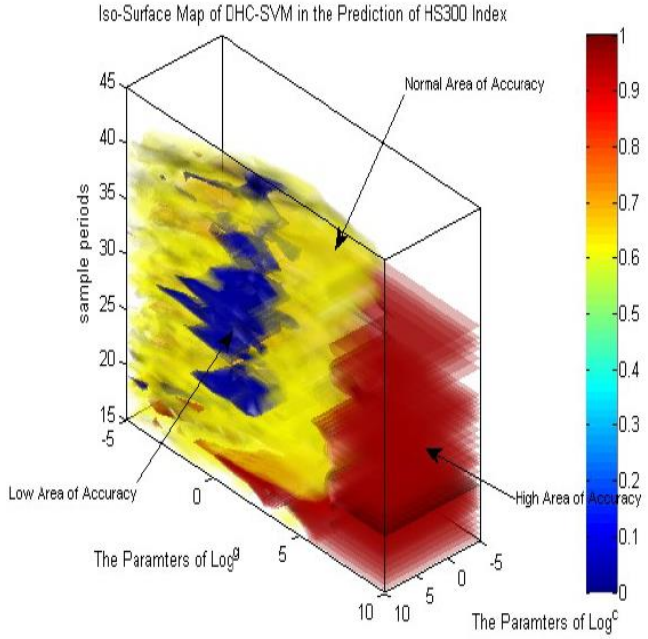

Figura 5. Resultados red SVM [24].

V. TIPOS DE ESCENARIOS

Se realizan pruebas en 4 escenarios de red para el IDS configurado con ANN. Las pruebas se realizaron durante las horas de trabajo. El primer escenario es una red inalámbrica en un segmento público. El segundo escenario es una red cableada en un entorno privado, principalmente con tráfico de World Wide Web. El tercer escenario es una granja de servidores con una configuración segura. El último escenario es una LAN en una zona militarizada (MZ) con VoIP y servicios de transmisión.

A. *Escenario Inalámbrico*

La arquitectura de la red inalámbrica Figura 6. Los servicios que se prestaran son el correo electrónico, la transmisión, la transmisión de datos y audio.

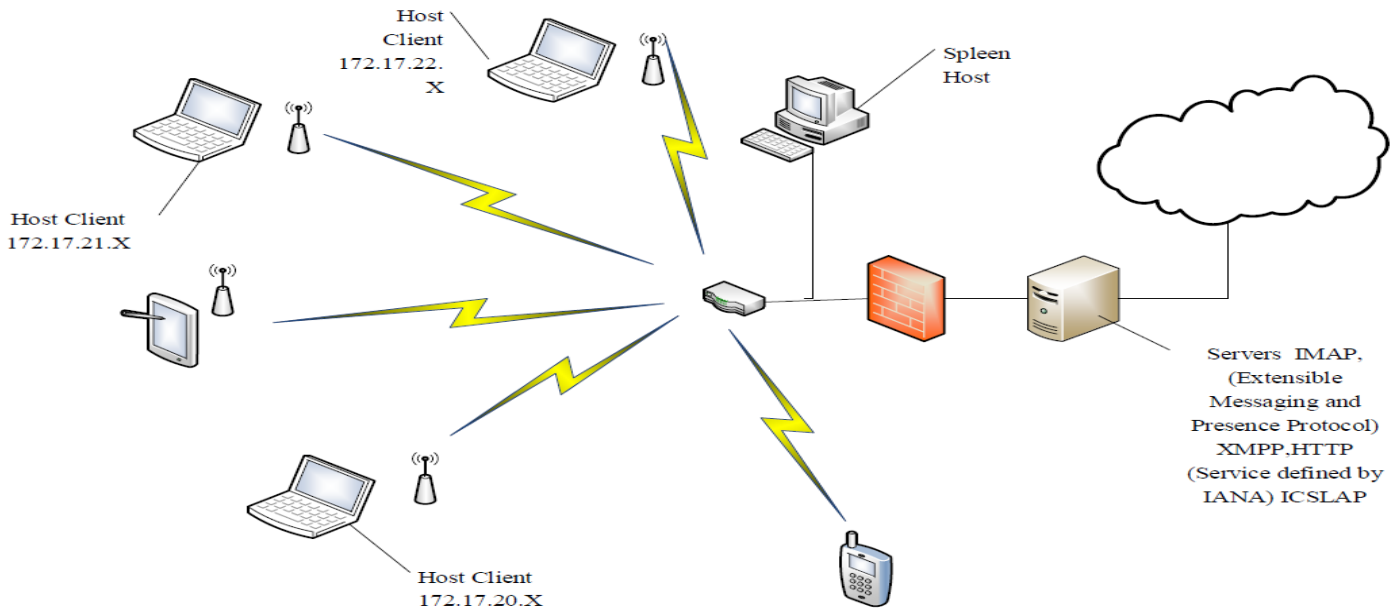

Figura 6 Arquitectura de red inalámbrica [1]

B. *Escenario Cableado*

La arquitectura de la red cableada Figura 7, esta red tendrá una estructura con 20 hosts conectados, incluidas las conexiones de puerto serie con dispositivos de salida estándar. El acceso del usuario está limitado a servicios web como correo electrónico corporativo, acceso a bases de datos y sitios web seguros.

C. *Granja de Servidores*

La red del servidor proporciona servicios, como firewall, DHCP local, transmisión básica, DNS y web servidores, streaming y FTP Figura 8.

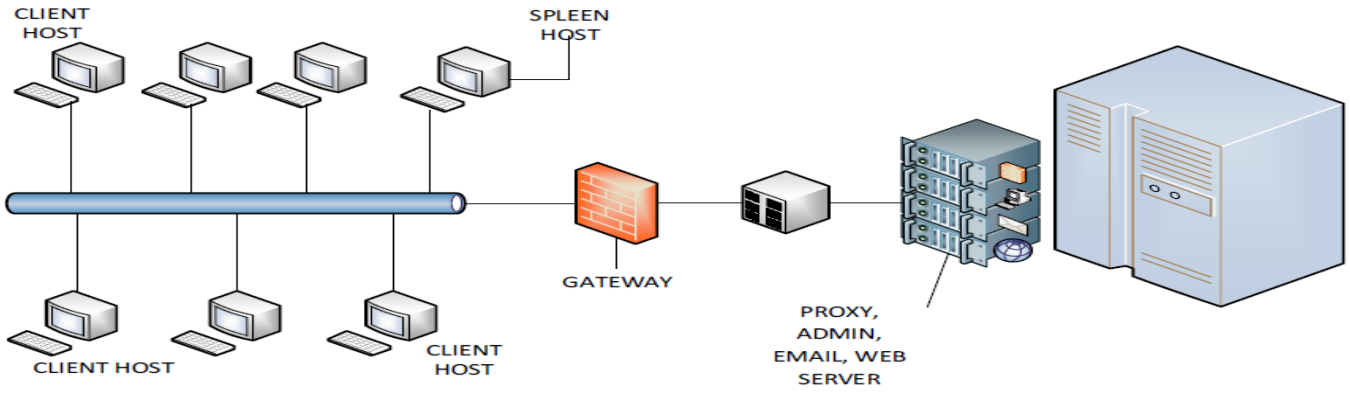

Figura 7 Escenario red cableada [1]

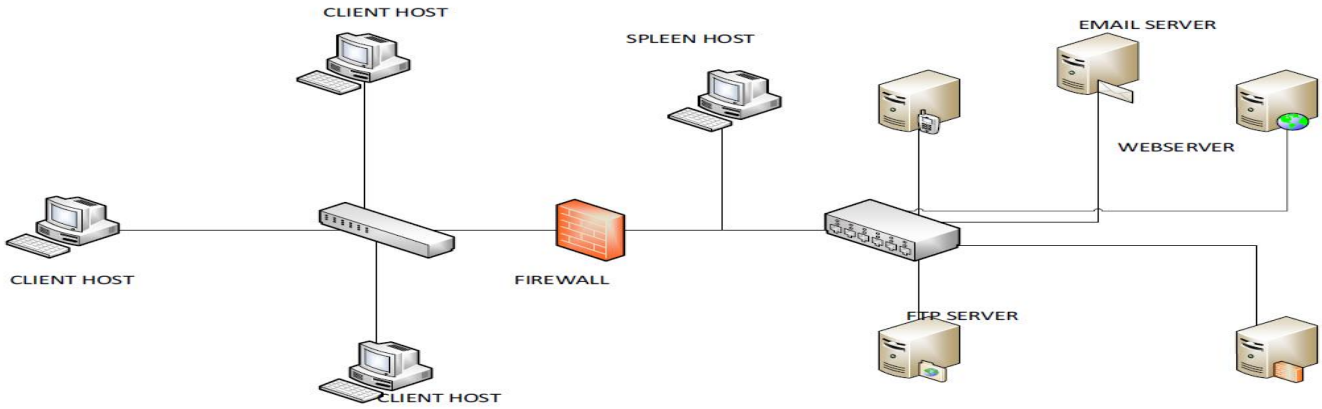

Figura 8 Escenario granja de servidores [1]

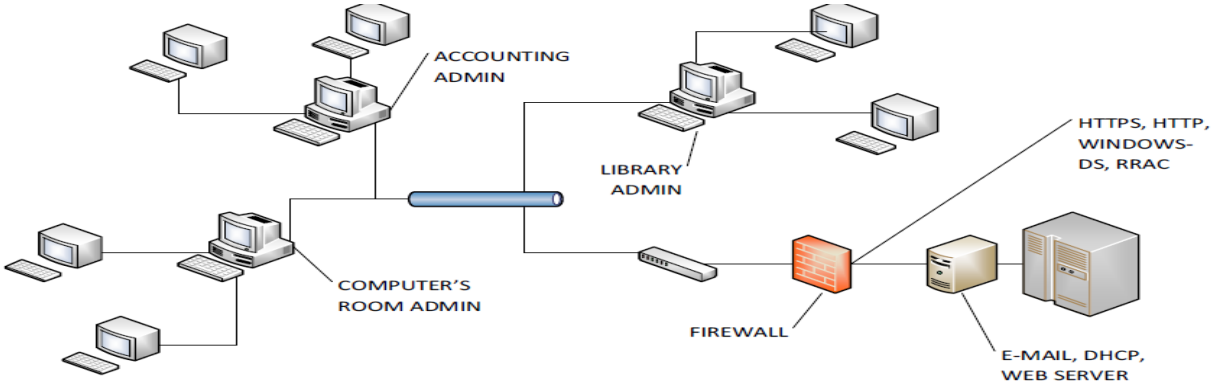

Figura 9 Escenario red militarizada [1]

D. LAN Militarizada

La LAN seleccionada está en una MZ Figura 9. Cada usuario tiene acceso a la intranet para actualizar y compartir información a través de los departamentos.

VI. DISCUSIÓN Y RESULTADOS

Los resultados obtenidos con la red neuronal se evaluaron con una tabla de verdad, la cual se define como: verdaderos positivos donde una intrusión es identificada correctamente, verdaderos negativos detectan un comportamiento inusual sin salir de los parámetros normales y sin ser intrusión, falsos positivos las anomalías sin ser intrusiones las detecta, y falsos negativos no detecta intrusos [14].

La ANN fue realizada con 2 capas ocultas cada una con 10 neuronas, la red fue escogida basándose en la Sección IV donde se presentaron algunos algoritmos de redes neuronales. Las ANN tienen más flexibilidad por la cantidad de capas ocultas que se pueden configurar con el fin de que si dos datos son muy cercanos tiene la posibilidad de separarlos en salidas diferentes, por otra parte el otro tipo de redes presentadas como SVM que realiza una clasificación para una predicción

sin adaptarse a nuevos datos y AG genera espacios para separar los diferentes comportamientos, en ambos casos los resultados de dos diferentes intrusiones que tengan valores muy parecidos, este tipo de redes puede llegar a generar una mayor cantidad de falsos positivos.

Para el entrenamiento de la ANN se usa la base de datos DARPA, como se mencionó en la Sección II, esta base de datos no solo contiene el comportamiento normal de la red si no también diferentes tipos de ataques a las redes de comunicaciones, con la cantidad de análisis recolectado es posible realizar entrenamiento de la red neuronal, validación y prueba de la red, verificando la efectividad en la detección de ataques.

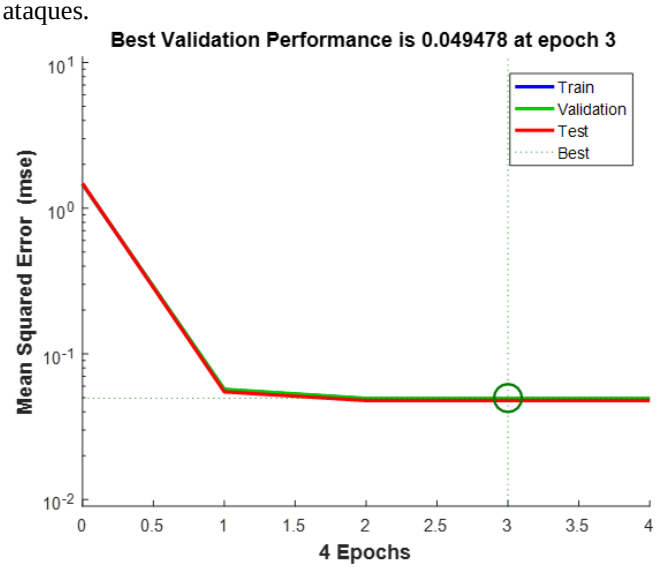

Figura 10 Representación gráfica de las mediciones de validación, entrenamiento y prueba del esquema propuesto

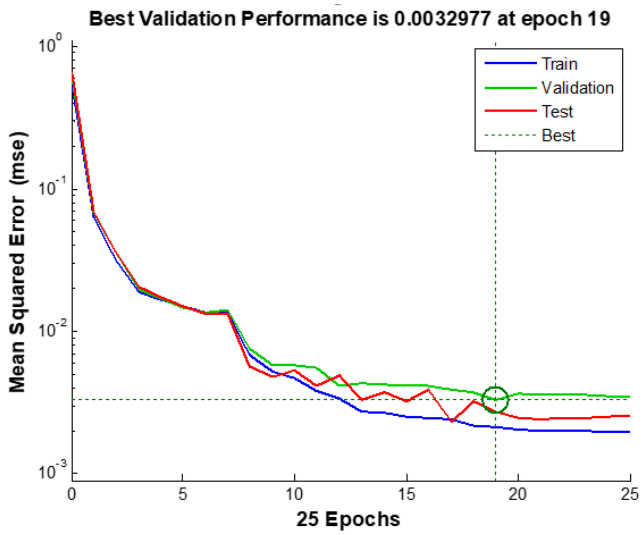

Figura 11 Representación gráfica de las mediciones de validación, entrenamiento y prueba del esquema desarrollado

Con el entrenamiento realizado se obtienen los siguientes resultados Figura 11 y Figura 10 red propuesta, donde la cantidad de épocas define la cantidad de veces que la prueba (datos que no hacen parte del entrenamiento) asemeja a la validación, acomodando los pesos de la red neuronal en el entrenamiento.

Los resultados observados Figura 12 y Figura 13, hay variación entre los escenarios propuestos, que como se ha mencionado hay diferentes tipos de servicios en cada uno y los errores de detección están por debajo del 25%, al comprar la red propuesta con la red [1], el MSE está mejor evaluado por la cantidad de datos que procesaron para entrenamiento y validación.

Red inalámbrica Hubo 100,023 conexiones durante el tiempo de prueba, lo que equivale a alrededor de 300 usuarios. Los conjuntos de datos se recolectaron en 51 buffers, y hubo 58,679 conexiones cerradas.

La arquitectura de red cableada hubo 35,047 conexiones. Los conjuntos de datos se recolectaron en 17 buffers, y hubo 27,465 conexiones cerradas con 3,534,549 paquetes.

La red del servidor usa uno de los protocolos principales para el tráfico analizado Kerberos [31]. Hubo alrededor de 100,100 conexiones. Los conjuntos de datos se recolectaron en 33 búferes, y hubo 3,839 conexiones cerradas con 6,769,309 paquetes. La zona desmilitarizada (DMZ) está compuesta por seis servidores, y también se incluye un host con la herramienta para monitorear el tráfico.

La LAN MZ y los servicios de gestión de tráfico son más grandes en la intranet que en las conexiones a Internet. Algunos servicios comunes son la conexión del agente de replicación remota (RRAC) con 5678 protocolos TCP / UDP. Otro servicio frecuente es Windows-DS a través del puerto 445. El número de conexiones fue de aproximadamente 111,000, con aproximadamente 15 usuarios durante el tiempo de prueba. Conjuntos de datos fueron recolectados en 21 buffers, y hubo 5,505 conexiones cerradas con 16,818,686 paquetes.

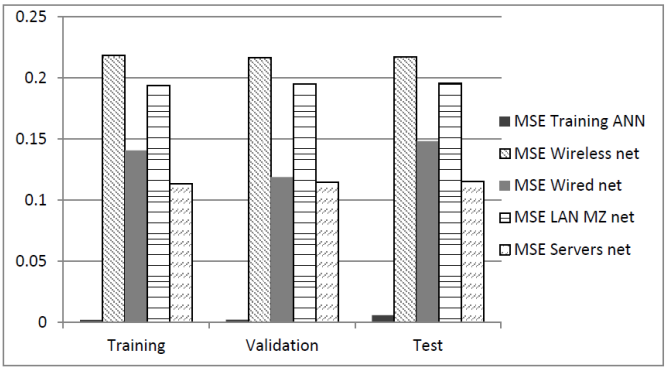

Figura 12 Pruebas realizadas en falsas detecciones en los diferentes escenarios [1]

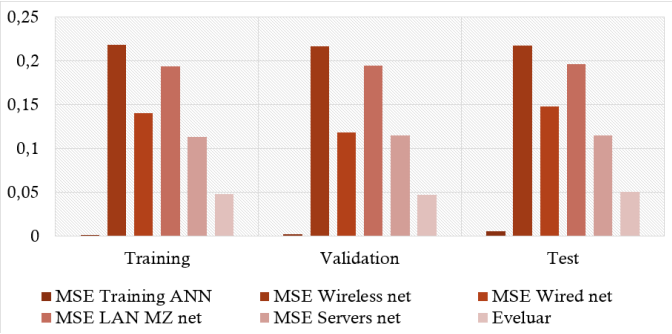

Figura 13 Pruebas realizadas en falsas detecciones en los diferentes escenarios con el algoritmo propuesto

VII. CONCLUSIONES

Como se esperaba para el comportamiento basado en AD, los detectores que fueron entrenados en un entorno de red que no son adecuados cuando se instalan en una red donde el comportamiento es muy diferente por eso la variación en los falsos positivos. Sin embargo, la ANN trata de modelar este tipo de variaciones y adaptarse a ellos manteniendo un error bajo.

Finalmente, los datos utilizados para este entrenamiento no son actualizados y aunque el sistema intente modelar los nuevos ataques, al no tener parámetros para actualizarse no le será posible detectar nuevas intrusiones por eso es recomendable usar bases de datos más actualizadas.

REFERENCIAS

[1] Guillen, E., Sánchez, J., & Paez, R. (2015). Inefficiency of ids static anomaly detectors in real-world networks. *Future Internet*, *7*(2), 94-109.

[2] W. Wang et al., "HAST-IDS: Learning Hierarchical Spatial-Temporal Features Using Deep Neural Networks to Improve Intrusion Detection," in IEEE Access, vol. 6, pp. 1792-1806, 2018. doi: 10.1109/ACCESS.2017.2780250 URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8171733&isnumber=8274985

[3] Fourati and K. Al Agha, "An IDS First Line of Defense for Ad Hoc Networks," 2007 IEEE Wireless Communications and Networking Conference, Kowloon, 2007, pp. 2619-2624. doi: 10.1109/WCNC.2007.487 URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4224732&isnumber=4224245

[4] Brown, M. Anwar and G. Dozier, "Intrusion Detection Using a Multiple-Detector Set Artificial Immune System," 2016 IEEE 17th International Conference on Information Reuse and Integration (IRI), Pittsburgh, PA, 2016, pp. 283-286. doi: 10.1109/IRI.2016.45 URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7785754&isnumber=7785148

[5] Axelsson, S. *Intrusion Detection Systems: A Survey and Taxonomy*; Technical Report: Chalmers University of Technology, Goteborg, Sweden, 14 March 2000.

[6] Liao, H.-J.; Richard Lin, C.-H.; Lin, Y.-C.; Tung, K.-Y. Intrusion detection system: A comprehensive review. *J. Netw. Comput. Appl.* **2013**, *36*, 16–24.

[7] Hu, W. Su, L. Wu, Y. Huang and S. Kuo, "Design of event-based Intrusion Detection System on OpenFlow Network," 2013 43rd Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Budapest, 2013, pp. 1-2. doi: 10.1109/DSN.2013.6575335

[8] Kayacik, H.G.; Zincir-Heywood, A.N.; Heywood, M.I. Selecting features for intrusion detection: A feature relevance analysis on KDD 99 intrusion detection datasets. In Proceedings of the Third Annual Conference on Privacy, Security and Trust, New Brunswick, Canada, 12–14 October 2005.

[9] Mukkamala, S.; Janoski, G.; Sung, A. Intrusion detection using neural networks and support vector machines. In Proceedings of the 2002 International Joint Conference on Neural Networks, 2002 (IJCNN'02.), Honolulu, HI, USA, 12–17 May 2002; pp. 1702–1707.

[10] Tang, H.; Cao, Z. Machine Learning-based Intrusion Detection Algorithms. *J. Comput. Inf. Syst.* **2009**, *5*, 1825–1831.

[11] A. Maske and T. J. Parvat, "Advanced anomaly intrusion detection technique for host based system using system call patterns," 2016 International Conference on Inventive Computation Technologies (ICICT), Coimbatore, 2016, pp. 1-4. doi: 10.1109/INVENTIVE.2016.7824846 URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7824846&isnumber=7824784

[12] Brugger, S. T., & Chow, J. (2007). "An assessment of the DARPA IDS Evaluation Dataset using Snort". UCDAVIS department of Computer Science, 1(2007), 22.

[13] Zhao, D.; Xu, Q.; Feng, Z. Analysis and Design for Intrusion Detection System Based on Data Mining. In Proceedings of the 2010 Second International Workshop on Education Technology and Computer Science (ETCS), Wuhan, China, 6–7 March 2010; pp. 339–342.

[14] Oluasola, A.A.; Oladele, A.S.; Abosede, D.O. Analysis of KDD 99 intrusion detection dataset for selection of relevance features. In Proceedings of the World Congress on Engineering and Computer Science, San Francisco, CA, USA, 20–22 October 2010; pp. 20–22.

[15] EDWARD PAUL GUILLEN PINTO, Spleen, Nombre comercial: Spleen, En: Colombia, 2012.

[16] Rodriguez, J. GTS: GNU Triangulated Surface Library. Available online: http://gts.sourceforge.net/ (accessed on 13 April 2015).

[17] Guillén, E.; Rodriguez, J.; Páez, R. Evaluating Performance of an Anomaly Detection Module with Artificial Neural Network Implementation. Int. J. Comput. Inf. Syst. Control Eng. 2013, 7, 836–842.

[18] Sommer, R.; Paxson, V. Outside the closed world: On using machine learning for network intrusion detection. In Proeedings of the 2010 IEEE Symposium on Security and Privacy (SP), Oakland, CA, USA, 16-19 May, 2010; pp. 305–316.

[19] Wang, K.; Stolfo, S.J. Anomalous payload-based network intrusion detection. In Proceedings of the Recent Advances in Intrusion Detection, Sophia Antipolis, France, 15–17 September 2004; pp. 203–222.

[20] Komyakov, A. A., Nikiforov, M. M., Erbes, V. V., Cheremisin, V. T., & Ivanchenko, V. I. (2016, June). Construction of electricity consumption mathematical models on railway transport used artificial neural network and fuzzy neural network. In Environment and Electrical Engineering (EEEIC), 2016 IEEE 16th International Conference on (pp. 1-4). IEEE.

[21] Sun, Y. Xu, G. Liang and Z. Zhou, "An Intrusion Detection Model for Wireless Sensor Networks With an Improved V-Detector Algorithm," in IEEE Sensors Journal, vol. 18, no. 5, pp. 1971-1984, 1 March1, 2018. doi: 10.1109/JSEN.2017.2787997 URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8241774&isnumber=8277082

[22] Liu, S., & Li, Z. (2017, February). A modified genetic algorithm for community detection in complex networks. In Algorithms, Methodology, Models and Applications in Emerging Technologies (ICAMMAET), 2017 International Conference on(pp. 1-3). IEEE.

[23] Tom Fawcett, "An introduction to ROC analysis", Pattern Recognition Letters, Volume 27, Issue 8, 2006, Pages 861-874, ISSN 0167-8655, https://doi.org/10.1016/j.patre .2005.10.010. (http://www.sciencedirect.com/science/article/pii/S016786550500303X)

[24] Xiaohua, S., & Yulin, Z. (2014, September). The implementation of dynamic heteroskedasticity convertible SVM model in financial time series. In Advanced Research and Technology in Industry Applications (WARTIA), 2014 IEEE Workshop on (pp. 281-285). IEEE.