Ciencia e Ingeniería Neogranadina
2023
Vol. 33(1)
enero-junio de 2023 ■ ISSN: 0124-8170 · ISSN-e: 1909-7735 ■ pp. 75-86

Editorial Neogranadina

DOI: https://doi.org/10.18359/rcin.6534

Sistemas de detección y prevención de intrusos: una taxonomía experimental basada en código abierto orientada a la industria 4.0*

Julio César Gómez Castaño[a] ■ Néstor Jaime Castaño Pérez[b] ■ Luis Carlos Correa Ortiz[c]

Resumen: este trabajo presenta una propuesta de taxonomía experimental basada en código abierto para los Intrusion Detection System/Intrusion Prevention System (IDS/IPS), orientada a la industria 4.0, debido a las necesidades actuales de seguridad de la información en hogares y empresas. Con la transformación digital, el crecimiento exponencial del Internet de las Cosas (IoT, por sus siglas en inglés), las conexiones a Internet y el aumento de amenazas, aumentan los problemas de seguridad de los equipos, que pueden verse vulnerados por los ciberdelincuentes y ser utilizados como intermedio para atacar otros equipos de la red propia, de otras organizaciones o para formar su propio botnet con miras a ataques masivos controlados. Por ello, es necesario contar con IDS/IPS que contribuyan a mejorar su seguridad. En la taxonomía se describe la infraestructura tecnológica en *hardware* y *software* para disponer en un ambiente experimental y realizar pruebas en la implementación, administración, gestión e investigación de IDS/IPS de código abierto y comprender las reglas y las anomalías para la detección de intrusos, mediante la base de datos de firmas y la utilización algoritmos de aprendizaje automático.

Palabras clave: IDS; IPS; open source; IoT; Machine Learning

Recibido: 23/11/2022

Aceptado: 30/ 03/2023

Disponible en línea: 07/07/2023

Cómo citar: J. C. Gómez Castaño, N. J. Castaño Pérez, y L. C. Correa Ortiz, «Sistemas de detección y prevención de intrusos: Una taxonomía experimental basada en código abierto orientada a la industria 4.0», Cien.Ing.Neogranadina, vol. 33, n.º 1, pp. 75–86, jun 2023.

* Artículo de reflexión.

a Especialista en Telecomunicaciones. Universidad de Manizales, Manizales, Colombia.
Correo electrónico: jgomez@umanizales.edu.co ORCID: http://orcid.org/0000-0002-0556-0758

b Doctor en Ingeniería. Universidad de Manizales, Manizales, Colombia.
Correo electrónico: ncastano@umanizales.edu.co ORCID: http://orcid.org/0000-0002-5450-6598

c Maestría en Ingeniería. Universidad de Manizales, Manizales, Colombia.
Correo electrónico: lcco@umanizales.edu.co ORCID: http://orcid.org/0000-0001-9488-5249

Intrusion Detection and Prevention Systems: an Open Source Based Experimental Taxonomy Oriented to Industry 4.0

Abstract: this paper presents a proposed open source-based experimental taxonomy for an Intrusion Detection System/Intrusion Prevention System (IDS/IPS) oriented to Industry 4.0 due to the current information security needs in homes and enterprises. With the digital transformation, the exponential growth of the Internet of Things (IoT), Internet connections, and the increase of threats, the security problems of the equipment increase, which can be vulnerable to cybercriminals and be used as an intermediary to attack other equipment of the own network, of other organizations or to form their botnet with a view to massive controlled attacks. Therefore, necessary to have IDS/IPS to help improve their security. The taxonomy describes the technological infrastructure in hardware and software to arrange in an experimental environment and perform tests in the implementation, administration, management, and research of open source IDS/IPS and understand the rules and anomalies for intrusion detection through the signature database and the use of machine learning algorithms.

Keywords: IDS; IPS; open source; IoT; Machine Learning.

Introducción

Con la cuarta revolución industrial, la humanidad avanza rápidamente hacia la transformación digital de los procesos, para que tanto en las empresas como en los hogares estos se puedan consultar, gestionar y administrar por Internet. Aunque se han incrementado los ataques cibernéticos es importante que quienes permanecen en línea lo hagan de forma segura, y empleen modelos que garanticen la protección de la infraestructura de red, los equipos y la información. De hecho, existe una necesidad creciente de poner a prueba ambientes de seguridad que detecten y corrijan errores o vulnerabilidades y que, además, incentiven la interacción con comunidades relacionadas con los sistemas de detección y prevención de aparición de intrusos.

En la actualidad existen muchas herramientas, aplicaciones y sistemas que ayudan a mejorar la seguridad de los dispositivos, programas e información que componen una red de datos, como por ejemplo, *firewalls*, redes privadas virtuales, antivirus y antimalware, criptografía, etc. Los IDS/IPS son herramientas que supervisan el movimiento de la red para detectar y bloquear tráfico malicioso y permitir solo el benigno y, de paso, mejorar la seguridad de la red, de los equipos y de la información.

Con el IoT y la cuarta revolución industrial, las empresas y los hogares tienen dispositivos, sistemas embebidos, cámaras Internet Protocol (IP), asistentes personales, etc., conectados en línea para prestar un mejor servicio. Algunos de ellos, al utilizar técnicas de inteligencia artificial (IA), se convierten en foco de ciberdelincuentes, que buscan acceder a su información sensible y creciente, aprovechando opciones como el aprendizaje automático (Machine Learning [ML]) para sus ataques.

Además de lo anterior, los dispositivos IoT presentan generalmente muchas debilidades en cuanto a la seguridad de la información desde su desarrollo, tales como: contraseñas débiles y almacenadas en texto plano, *firmware* desactualizado y errores en su desarrollo, y comunicación entre dispositivos y servidores sin cifrar. Por ello, los delincuentes informáticos están explotando esas vulnerabilidades con herramientas actuales y técnicas propias como *Autosplit*, para lograr su objetivo [1].

En el mercado existen diversas soluciones de IDS/IPS, tanto comerciales (Fortinet, Cisco, Palo Alto, que se basan en *hardware* integrado con el *firewall*), como basadas en *software* libre (Snort, Suricata, Zeek). En el presente trabajo se enfatizará en las herramientas libres asequibles, que ofrecen beneficios a expertos en seguridad, hogares y organizaciones del ámbito regional.

En este artículo, a partir de la experiencia de sus autores, se recomienda la infraestructura tecnológica en *software*, *hardware* y conjunto de datos (dataset) para un ambiente de pruebas basado en *software* libre orientado a implementar, administrar y gestionar IDS/IPS con la taxonomía propuesta. Además, se consideran algunos algoritmos de Machine Learning (ML) utilizables en la clasificación de las anomalías, especialmente ataques desconocidos, que permiten obtener análisis comparativos con ataques previamente identificados en el proceso de aprendizaje.

También se recomendarán las pruebas con los algoritmos de clasificación de ML para detectar anomalías desconocidas, con base en el entrenamiento y tests con los dataset seleccionados y finalmente se mostrarán los experimentos iniciales realizados en los IDS/IPS.

Sistemas de detección y prevención de intrusos

Un sistema de detección de intrusos (*Intrusion Detection System* [IDS]) es una aplicación o *hardware* que supervisa el tráfico que circula por una red para detectar actividades sospechosas que supongan una amenaza. Si el sistema tiene la capacidad de bloquear el tráfico proveniente del origen del paquete con la información maligna, se llama sistema de prevención de Intrusos (*Intrusion Prevention System* [IPS]); dependiendo de la ubicación del IDS/IPS en la topología de la red, se puede monitorear el tráfico de la red o de la subred o de un solo equipo [2].

En los trabajos [3, 4] explican que hay seis módulos en la solución de IDS, tal como se observa en la figura 1. Ellos son: 1) Módulo de captura: se

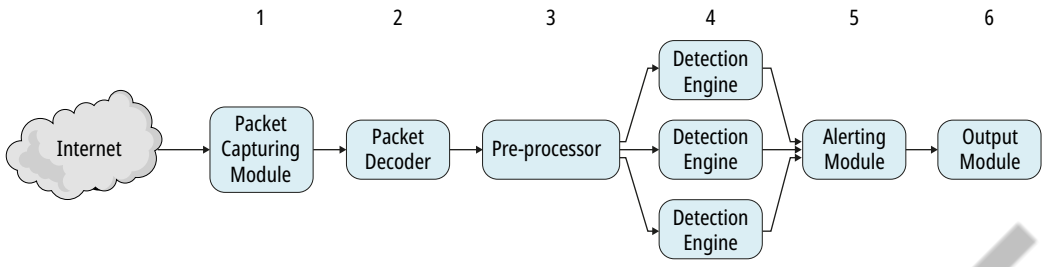

Figura 1. Representación general por bloques del IDS.
Fuente: Waleed, 2022 [4, p. 2].

utiliza para adquirir datos desde la red. Para su implementación se utiliza normalmente Libpcap y AF_Packets; 2) Módulo decodificador: decodifica la información para determinar las características básicas de la red, como el origen y destino de los datos y las direcciones de los puertos, entre otros; 3) un preprocesador ensambla los paquetes, si están fragmentados, y analiza y normaliza los datos, que usarán más adelante los protocolos de la capa de aplicación (HTTP, FTP, DNS, etc.); 4) Módulo motor de detección: es la parte crucial de los IDS, donde las reglas para detectar ataques se comparan con las entradas; 5) Módulo de alertas: se presenta cuando una regla coincide con los atributos del paquete; el IDS muestra mensajes informativos, mientras el IPS bloquea el tráfico y genera la alerta y 6) Módulo de salida: refleja las estadísticas de la detección del IDS/IPS.

Taxonomía experimental de los IDS/IPS

En la figura 2 se observa la propuesta de la taxonomía experimental basada en código abierto para la realización de pruebas de laboratorio, que le permitirá adquirir la habilidad para implementar, administrar, gestionar e investigar en los IDS e IPS y solucionar problemas de seguridad de la

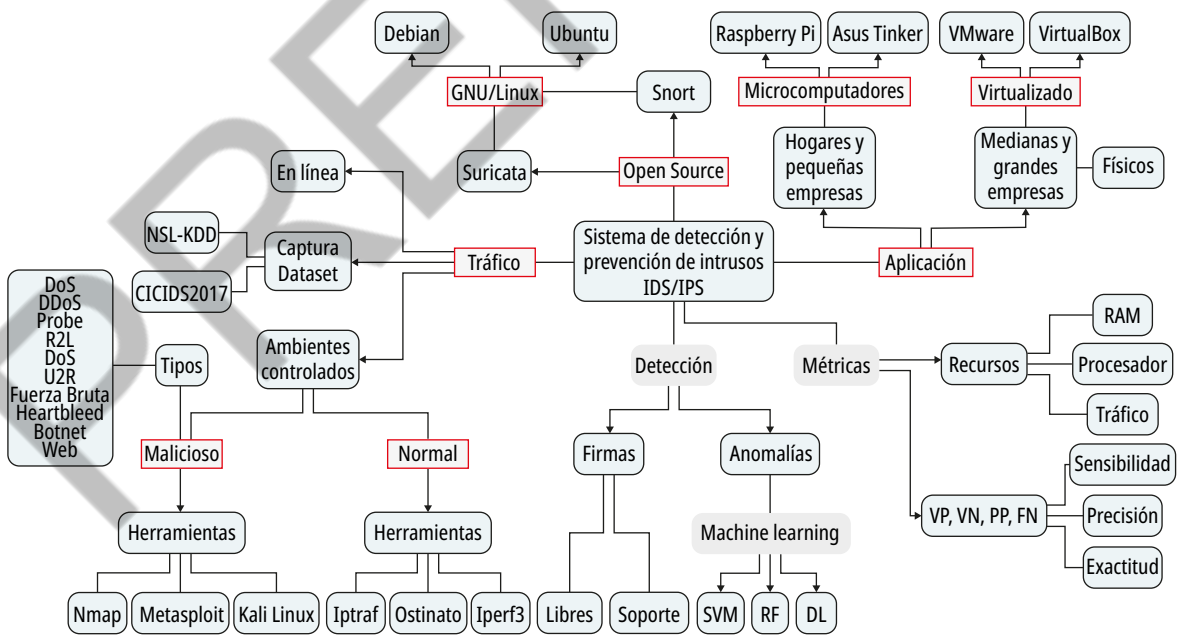

Figura 2. Propuesta de la taxonomía experimental basada en código abierto para IDS/IPS
Fuente: elaboración propia.

información en hogares y organizaciones. En el primer nivel de la clasificación se encuentran la aplicación de los experimentos, las soluciones de *software* libre que se utilizan para la detección y prevención de intrusos, las métricas para las mediciones y comparaciones de desempeño, las formas de detección de tráfico malicioso y la fuente de donde se obtendrán los datos para el análisis.

Aplicación de los experimentos

Los experimentos que se realizarán en el laboratorio de seguridad y detección se enfocarán en la necesidad de brindar una mejor protección a los dispositivos conectados a Internet y a la información de hogares y pequeñas empresas y en las medianas y grandes organizaciones.

Hogares y pequeñas empresas: como estas instituciones no requieren una gran infraestructura tecnológica, su conexión a Internet es a baja velocidad; se les debe brindar una solución de IDS/IPS de bajo costo, acorde con sus capacidades y necesidades; en esta categoría encontramos hogares y empresas que tienen menos de 30 dispositivos entre computadores, teléfonos inteligentes, asistentes personales, cámaras IP y dispositivos IoT.

Medianas y grandes empresas: estas organizaciones que cuentan con una mayor infraestructura tecnológica, conexiones dedicadas y mayores anchos de banda de conexión a Internet, con servidores públicos y privados, pueden contar con un equipo de cómputo o un sistema virtualizado para la implementación de un IDS/IPS basados en código libre.

Soluciones de código abierto para IDS/IPS

Para el manejo de la detección y prevención de intrusos se necesita un *software* que realice este proceso. Uno de los IDS/IPS de código abierto más reconocidos en la actualidad es Snort, respaldado por la empresa Cisco; provee la base de datos de las reglas para la detección de tráfico malicioso, está disponible de forma libre y lo pueden usar otros IDS/IPS. Se ha convertido en un sistema estándar de firma de patrones en los IDS.

Por otra parte, Suricata es un motor de detección de código abierto, que puede actuar tanto de IDS como de IPS. El proyecto lo sostiene la Open Information Security Foundation (OISF), lo que la convierte en una opción muy fiable, ya que cuenta con el respaldo de la comunidad; prueba de ello es la frecuencia con la que se realizan mejoras y actualizaciones al código fuente.

Métricas en IDS

Para la evaluación del desempeño de un IDS/IPS se debe comparar el consumo de los recursos de RAM, procesador, la pérdida de paquetes y los tiempos de respuesta de las diferentes opciones de *software* libre para la detección de posibles ataques cibernéticos. Igualmente se han identificado cuatro métricas asociadas a la naturaleza del evento y el estado de la detección. Esas métricas son verdadero positivo (VP), un ataque correctamente identificado como maligno, verdadero negativo (VN), un tráfico normal correctamente reconocido como tráfico benigno, falso positivo (FP), un tráfico normal identificado incorrectamente como ataque y falso negativo (FN), un ataque identificado incorrectamente como tráfico normal. Para la comparación del desempeño de los diferentes algoritmos de clasificación de los IDS que determinan si un paquete es tráfico normal o un ataque se dispone de las siguientes métricas: [5, p. 11].

- Exactitud (*Accuracy*): se define como el porcentaje de predicciones correctas, es decir, el porcentaje de tráfico anómalo que se clasifica correctamente. Es la relación entre las detecciones correctas y el número total de registros en el conjunto de datos. Se calcula de la siguiente manera:

$$Exactitud = \frac{VP + VN}{VP + FP + FN + VN} \quad (1)$$

- Precisión (*Precision*): esta métrica describe la capacidad del clasificador para predecir datos normales sin condiciones. Define la proporción de verdaderos positivos contra todos los resultados positivos.

$$Precisión = \frac{VP}{VP + FP} \quad (2)$$

- Sensibilidad (*Recall*): es la relación entre el número de registros clasificados correctamente

y el número de todos los eventos correctos. Se calcula así:

$$Sensibilidad = \frac{VP}{VP + FN} \quad (3)$$

Detección de intrusos

En el módulo de motor de localización existen dos estrategias para la detección de intrusos: una basada en firmas (*signature detection*) que consiste en descubrir intrusos a partir de una base de datos de firmas o indicadores creada desde ataques anteriores. La otra estrategia se conoce como detección de anomalías (*anomaly detection*) que se logra con base en el monitoreo, recolección y análisis de los paquetes de la red para un posterior análisis y clasificación de un comportamiento normal y uno anómalo. La detección de anomalías puede plantearse como un problema de clasificación con ML, que es una familia de algoritmos de la IA capaz de aprender, sin estar explícitamente programada, gracias a un conjunto masivo de datos (dataset), que le posibilitan a una red de trabajo mantenerse alerta sobre las posibles amenazas. Posteriormente, y gracias a los nuevos datos que procesa, esta máquina se va entrenando más y más para mejorar su clasificación [6].

- **Detección basada en firmas:** en este módulo el sistema más utilizado por los IDS son las reglas del IDS/IPS Snort de libre acceso. Los usuarios añaden firmas de ataques de forma manual. En el trabajo [7] utilizan técnicas de ML para encontrar anomalías en el tráfico y generar reglas automáticas para el IDS. Para definir reglas y comprender el funcionamiento es fundamental conocer su sintaxis y lo que representan los atributos: [8, p.13]
  - [acción] [protocolo] [IP origen] [puerto origen] -> [IP destino] [puerto destino] ([Opciones de regla])
- **Acción:** indica qué debe hacer el IDS cuando un paquete coincida con el criterio de una regla. Las posibles opciones son: alert, log, pass, drop, reject y sdrop.
- **Protocolo:** tipo de protocolo del modelo TCP/IP, las opciones son: IP, ICMP, TCP, UDP, HTTP, etc.
- **IP origen y destino:** dirección (es) IP de origen/destino del paquete, que pueden configurarse con la palabra clave *any*, que representa cualquier dirección IP.
- **Puerto origen/destino:** el puerto o los puertos de origen/destino; este atributo cuenta con la posibilidad de configurarse con el comodín *any*.
- **Operador de dirección:** indica la dirección del tráfico que se aplica a la regla; las dos opciones son: -> o <-
- **Opciones de regla:** las opciones forman la base del motor de detección del Snort, incluyen diversas alternativas que se separan por una coma y se dividen en cuatro grandes categorías: las generales, *payload*, *non-payload* y las *post-detection*. Ejemplo de una regla básica de Snort:
  - alert tcp any any -> 192.168.23.2 80 (msg: "Prueba detección http"; content: "Es consulta http" ; sid 1);

La regla tiene activada la acción de *alert*. En caso de coincidencia generará una alerta. Además, detectará segmentos TCP que se dirijan a la dirección IP privada 192.168.23.2, desde cualquier dirección IP origen y desde cualquier puerto al 80, que identifica el servicio HTTP. En las opciones de la regla se especifican varias alternativas generales como sid, es decir, el identificador único de la regla, y msg y content, cadenas de texto descriptivas que proporcionan información sobre la alerta [8, p. 13].

Machine Learning es una forma de la IA que enseña a la computadora cómo revelar patrones y hacer conexiones entrenando con un gran volumen de datos. La máquina utiliza una gran cantidad de datos y algoritmos sofisticados para saber cómo realizar la tarea por sí misma [9, p. 13].

Tráfico para analizar

Para la captura de paquetes que van a analizar los IDS/IPS en la taxonomía experimental, se definieron las siguientes tres categorías: tráfico en línea, captura de los dataset y ambientes controlados.

- **Tráfico en línea:** en la presente clasificación los IDS/IPS se conectan a la red de datos de la

organización por un tiempo limitado y se comparan sus resultados tal como se hizo en el artículo [10] donde se analizó el tráfico por cinco meses para cotejar los resultados en el bloqueo de direcciones IP origen del Snort con el de Suricata en la Universidad de Londres.

- **Captura de los dataset:** se utilizarán en los laboratorios de pruebas, para el entrenamiento y test de la clasificación con base en los algoritmos de ML y comparación de desempeño; los dos conjuntos de datos más utilizados y actualizados según la revisión bibliográfica son: NSL-KDD, que contiene las firmas de ciberataques de cuatro tipos y consta de 148.527 registros, está formada por 41 características [5, p. 5] y CICIDS2017, que consta de tráfico de red normal de los siguientes protocolos: HTTP, HTTPS, FTP, SSH y protocolos de correo y de tráfico de ataques de siete tipos. El conjunto de datos tiene en total 2.973.635 registros y 69 características [7-11].
- **Tráfico generado en ambientes controlados:** es otra categoría para generar tráfico en una red independiente en hogares y empresas y utilizar herramientas para generar el tráfico normal y malicioso que generan los usuarios y los delincuentes informáticos. Dentro de los posibles ataques que pueden realizar y que están definidos en la taxonomía propuesta son [7-11]:
  - DoS: denegación de servicios (*Denial of Service* [DoS]). El atacante intenta que el servicio no esté disponible para los usuarios legítimos mediante la carga o inundación de enormes paquetes no deseados.
  - DDoS: denegación de servicios distribuido (*Distributed Denial of Service* [DDoS]), cuando varios equipos realizan el ataque de DoS para consumir el ancho de banda o los recursos de la víctima (RAM, procesador, disco duro).
  - *Probe*: el ataque de sondeo, el atacante monitorea a la víctima remota y trata de recolectar alguna información mediante el escaneo de puertos para determinar los servicios que tiene activos el sistema de cómputo o el sistema operativo que utiliza, entre otros.
  - R2L: buscando acceso local sin cuenta (*Remote to Local* [R2L]). En esta forma de ataque, el intruso intenta acceder al sistema sin contar con las credenciales para ello.
  - U2R: elevación de privilegios (*User to Root* [U2R]). El atacante ya tiene acceso local a la máquina de la víctima e intenta obtener mayores privilegios.
  - Fuerza bruta: es de los ataques más populares que se pueden utilizar para descifrar contraseñas y para encontrar páginas y contenidos ocultos en los servidores web; utiliza todas las combinaciones posibles para lograrlo.
  - *Heartbleed*: el ataque sangrado del corazón proviene de un fallo en la biblioteca criptográfica OpenSSL. Normalmente se explota enviando una solicitud de *heartbeat* malformada con una carga útil pequeña y un campo de longitud grande a la parte vulnerable del servidor para obtener acceso a la información cifrada.
  - *Botnet*: el ataque red zombi es una serie de dispositivos infectados conectados a Internet (computadores, celulares inteligentes, IoT) utilizados por el propietario de una botnet para realizar diversos ataques. Puede utilizarse para robar datos, enviar *spam* y permitir al atacante el acceso al dispositivo y a su conexión.
  - Web: el ataque web emplea la inyección Structured Query Language (SQL), que es una cadena de comandos SQL y la inyección de secuencia de comandos en sitios cruzados (*Cross-Site Scripting* [XSS]) para tener acceso a la base de datos. Además, permite a los atacantes implantar *scripts* maliciosos en el sitio web legítimo.
  - Infiltración: la infiltración en la red desde el interior se realiza explotando un *software* vulnerable como Adobe Acrobat Reader. Una vez explotado con éxito, se ejecutará una puerta trasera en el computador de la víctima y se realizarán diferentes ataques en la red, por ejemplo, el escaneo de direcciones IP y la enumeración de servicios mediante la herramienta libre Nmap.

Experimentos por realizar

Con la ayuda de la taxonomía propuesta se van a realizar experimentos prácticos para probar la confidencialidad, integridad y disponibilidad de la información en hogares y empresas con base en pruebas de penetración éticas, las comparaciones de desempeño de los recursos donde están soportados los IDS/IPS y la eficiencia, precisión y sensibilidad en la detección de intrusos tanto basados en firmas como en anomalías.

Pruebas de penetración éticas

En los ambientes controlados se realizan pruebas de penetración éticas a la infraestructura de red con la que normalmente cuentan los hogares y las empresas, con las herramientas propuestas, para después verificar que con la puesta a punto de los IDS/IPS, se detecta el ataque malicioso y se bloquean los ataques. Como se evidencia en el trabajo de [12] que implementaron el IDS/IPS Suricata para proteger un lector Radio Frequency Identification (RFID) y utilizaron Raspberry Pi. El *firewall* IPtables se configura para pasar la comunicación al Suricata y la herramienta *port knocking* para activar la conexión desde determinadas direcciones IP, ya que el lector *Low Level Reader Protocol* (*LLRP*) no solicita autenticación. También usaron las reglas del IDS/IPS para la autorización de lectura o administración. Al lector RFID protegido con el Suricata le realizaron tres pruebas de penetración y el sistema quedó bien protegido, gracias a que la herramienta *port knocking* puede activar/desactivar reglas de *firewall* de forma remota con base en una secuencia de puertos TCP.

Comparaciones de desempeño

Para probar el funcionamiento de los IDS/IPS y para las comparaciones tanto en el consumo de recursos (RAM, procesador, pérdida de paquete, latencia) y el desempeño de métricas de exactitud, precisión y sensibilidad en la detección de intrusos. En la revisión bibliográfica se destacan los siguientes trabajos:

En la investigación [13] compara el rendimiento de los IDS Snort y Suricata, en un ambiente controlado y virtualizado utilizando VirtualBox con dos máquinas iguales con sistema Operativo CentOS para cada uno de los IDS, un servidor Web con los servicios de SSH y FTP como el objetivo de ataque de las pruebas, otro servidor con las herramientas libres Ostinato, NMAP, HPING para generar tráfico normal y un quinto servidor para generar tráfico malicioso, utilizando el *framework* de Metasploit y un *suiche* virtualizado.

En [4] compararon tres herramientas libres para IDS (Snort, Suricata y Zeek) y realizaron pruebas de desempeño y en [3] compara el desempeño de Snort y Suricata en redes de alta velocidad (10 Gbps y 100 Gbps) utilizan Iperf3 para generar el tráfico.

En [14, p. 30] utilizan suiches Ciscos y Siemens, con dos IDS/IPS y Kali Linux; el Snort y el Suricata se implementan sobre servidores GNU/Linux como Debian, Ubuntu y CentOS. En la figura 3 se advierte el ejemplo de la topología que implementaron en VMware Workstation Pro (versión 15.5), los IDS/IPS Suricata (versión 5.0.0), y Snort (versión 2.9.16 y 3.1.7), instalados sobre el sistema operativo Ubuntu 18.04. Cada máquina está configurada con cuatro CPU cores y 16 GB de RAM, el generador de tráfico corre sobre Windows 10 y el equipo objetivo está sobre Ubuntu 18.04; las dos máquinas están configuradas con dos CPU cores y 4 GB de RAM. El suiche está configurado con puerto espejo (*port mirror*) para que el tráfico que vaya hacia el equipo objetivo, lo dirija también a los tres puertos del suiche donde están los IDS. En el trabajo de [15] utilizaron suiches virtuales con la función de

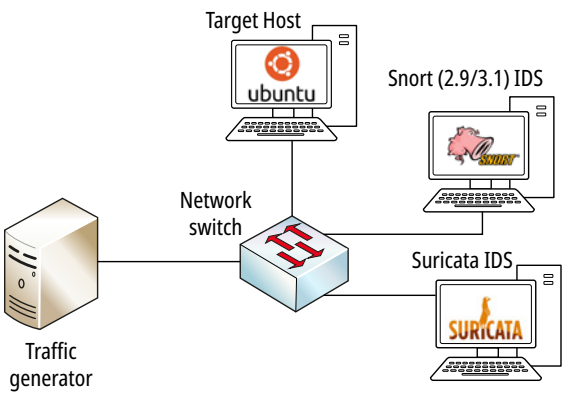

Figura 3. Topología de prueba de IDS
Fuente: Waleed, 2022 [4, p. 2]

puertos espejos para la retransmisión de paquetes al servidor con el Suricata.

En la investigación [16] utiliza Raspberry para la implementación del IDS/IPS. El costo del equipo es muy inferior al de otras soluciones y tiene una gran aceptación entre la comunidad, debido a su accesibilidad, modularidad y rendimiento. El modelo emplea una Raspberry Pi 3 B (cuyo costo asciende a 43 USD) y el adaptador USB Ethernet (que cuesta 8 USD). Como se muestra en la figura 4, utilizaron las reglas del Snort con pago de suscripción.

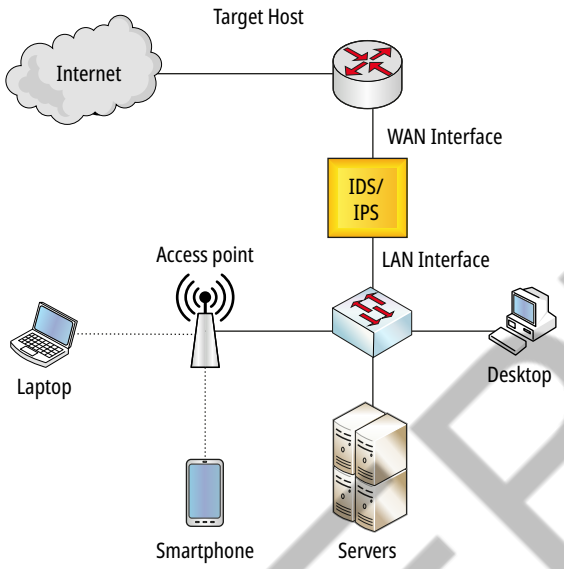

Figura 4. Raspberry Pi como un IDS/IPS
Fuente: De la Cruz, 2018 [16, p. 5]

Experimentos en la detección

Para entender cómo funciona la detección de trafico malicioso se debe conocer la sintaxis de la firma de los ataques conocidos y crear nuevas o tener reglas personalizadas acordes con las necesidades de los hogares y las empresas. Se deben realizar experimentos para detectar ataques nuevos por medio de técnicas de ML con algoritmos de clasificación para determinar si un paquete es benigno o maligno y evitar problemas de amenazas *Zero days* (no hay reglas o firmas en la base de datos para detectar esos nuevos ataques) y también se recomienda comparar con los principales algoritmos de clasificación utilizando las métricas de exactitud, precisión y sensibilidad. Las investigaciones más significativas son:

En su trabajo, [17] comparan algoritmos conocidos de aprendizaje automático en la clasificación como: máquinas de vectores de soporte (*Support Vector Machine* [SVM]), bosque aleatorio (*Random Forest* [RF]) y *Extreme Learning Machine* (ELM). En la investigación [13] en la comparación de Snort y Suricata detectaron una alta tasa de FP, por lo cual desarrollaron un *plugin* para Snort, utilizando el *software* de minería de datos Weka para el preprocesamiento de datos con cinco algoritmos de ML de alto rendimiento: SVM, árboles de decisión (*Decision Trees* [DT]), lógica difusa, *BayesNet* y *Naive-Bayes*. En el artículo [5] para la clasificación como tráfico normal o anormal de diferentes categorías de ataques (DoS, U2R, R2L, Probe) para IDS y la utilización de conocimiento profundo (*Deep Learning* [DL]), en una red con dispositivos IoT.

Recomendaciones de la infraestructura tecnológica

En la taxonomía experimental propuesta basada en código abierto para la realización de pruebas en laboratorio para IDS/IPS se encuentran las recomendaciones de la infraestructura tecnológica.

Aplicación de los experimentos

Los experimentos que se van a realizar se enfocan en las necesidades de seguridad de la información en hogares y empresas; adicionalmente, se recomienda el equipo necesario para contar con IDS/IPS en la infraestructura de red y el manual para la implementación.

- Infraestructura para hogares y pequeñas empresas: estas organizaciones no requieren un computador para implementar el IDS/IPS, ya que pueden usar una microcomputadora [18]; es un dispositivo pequeño, del tamaño de una tarjeta de crédito con un microprocesador como su unidad central de procesamiento (CPU, por sus siglas en inglés). Generalmente en el microprocesador los circuitos de almacenamiento y entrada/salida están en el mismo circuito integrado. Estos equipos son muy utilizados para

IoT, como: servidores web o de impresión, para video juegos, etc. Los más conocidos son [18]: Raspberry Pi, ASUS Tinker, LePotato, La Frite y otros. En los artículos analizados para IDS/IPS todos utilizan Raspberry Pi, el dispositivo para implementar el IDS/IPS en esta clasificación. En [18] explican la instalación y configuración del IDS/IPS Suricata en una Raspberry Pi y el material necesario.

- Infraestructura para medianas y grandes empresas: para contar con laboratorios de pruebas para la implementación de IDS/IPS libre para medianas y grandes empresas, se requieren computadores físicos o virtuales. En el laboratorio que realizaron en la investigación [4] se necesitaron cinco computadores (IDS/IPS_1, IDS/IPS_2, generador de tráfico benigno, generador de trafico malicioso y el equipo objetivo de los ataques) con al menos 16 GB de RAM y un procesador Core I5 o I7. Si se usa una máquina virtual se debe de contar con al menos 64 gigas de RAM.

Código abierto para los IDS/IPS

Para el manejo de la detección y prevención de intrusos se necesita del *software* que realice esta función; con base en la búsqueda bibliográfica de IDS/IPS de código abierto, donde siempre aparecen las herramientas Snort o Suricata y en la experiencia profesional de 26 años, trabajando en el *software* libre, estos son los IDS/IPS más utilizados en la actualidad, es decir, que se recomiendan para los diferentes experimentos en los ambientes de pruebas, tanto para los hogares (sobre las Raspberry) como para las empresas. Por lo demás, se instalan sobre los sistemas operativos libres: GNU/Linux Ubuntu, Debian y CentOS.

Detección por firmas y anomalías

Para este ambiente de pruebas se recomienda recurrir a la base de datos de firmas de ataques conocidos del Snort que son libres y se pueden descargar de su página oficial [20]. El mismo Snort ofrece la opción de la versión con suscripción; las reglas están disponibles desde su lanzamiento, mientras que los otros deben esperar 30 días para su liberación. El costo anual individual y para hogares es de 29.99 USD y para empresas es de 399 USD por cada sensor [21]. La sugerencia para las pruebas de hogares y personas es adquirir la licencia. De todas formas, en los laboratorios de pruebas se van a utilizar los dos tipos de reglas.

Para la detección por anomalías los algoritmos son: SVM, RF y DL. Se utilizará la misma infraestructura tecnológica descrita antes y los mismos dos conjuntos de datos (NSL-KDD y CICIDS2017) para el entrenamiento y test y los programas de Python, R y WEKA para las pruebas de los algoritmos de clasificación.

Generación de tráfico

Para la generación de tráfico normal (benigno), con base en el estado del arte y la experiencia de los autores en *software* libre, se recomiendan las siguientes herramientas: IPTtraf, ostinato e iPerf3 y para la generación de tráfico malicioso y ataques las siguientes herramientas: Nmap, Metasploit y Kali Linux. De igual forma, se van a manejar las capturas de tráfico cuando se realiza el proceso de la generación de los datasets para los test de evaluación de los IDS/IPS; los dos conjuntos de datos más reconocidos y actualizados, según la revisión bibliográfica son: NSL-KDD y CICIDS2017, seleccionados en la taxonomía propuesta.

Recomendaciones de la infraestructura tecnológica

En la actualidad se han realizado pruebas de funcionamiento del IDS/IPS Suricata (versión 6.0.4.1) instalado sobre el *firewall* Pfsense en las versiones 2.4.5 y 2.6.0 (última versión), que aplican las reglas del Snort, sobre un Acer Aspire 3 con procesador Core I3 de 7th Generación, 12 GB de RAM y disco duro de estado sólido, en un sistema operativo Windows 10 con VirtualBox (versión 6.3.1), para la instalación de la máquina virtual Kali Linux (versión 2022). Para la generación de tráfico malicioso se hicieron pruebas de Probe y el IDS/IPS detectó el ataque y bloqueó el tráfico proveniente del Kali Linux; el computador Acer respondió bien a estas pruebas iniciales. La instalación y configuración se realizó con base en las guías [20-22].

Conclusiones

Se creó una taxonomía experimental basada en código abierto para la implementación de unos ambientes de prueba de IDS/IPS, indicando que para su puesta en funcionamiento y para mejorar la seguridad de la información en los hogares y empresas se requiere infraestructura en *hardware* y *software*.

Los primeros experimentos con los IDS/IPS se pueden hacer con computadores o microcomputadores, cuyos precios son asequibles y los programas y reglas son de acceso libre. De este modo es fácil familiarizarse con el manejo de estas herramientas de seguridad y bajar los costos en hogares y empresas que demandan estos sistemas de seguridad.

Con IDS/IPS y herramientas libres como *port knocking* y el *firewall* IPtables, se mejora la seguridad en dispositivos IoT y otros equipos, ya que, al no requerir autenticación para modificar la configuración, un servicio que tiene debilidades de seguridad se puede controlar con estas herramientas, como se aprecia en el trabajo [12].

Disponer de la infraestructura de la taxonomía experimental de IDS/IPS, permite la ejecución de pruebas de penetración ética en los ambientes controlados, el manejo de la detección de intrusos por firmas y anomalías con la utilización de algoritmos de clasificación de *Machine Learning* e interactuar con comunidades del código libre y de detección de intrusos beneficia la región y el país.

Referencias

[1] J. Ruiz *et al.*, "How to Improve the IoT Security Implementing IDS/IPS Tool Using Raspberry Pi 3B+", *International Journal of Advanced Computer Science and Applications*, vol. 10, n.° 9, The Science and Information Organization, 2019, pp. 399-405, DOI: https://doi.org/10.14569/ijacsa.2019.0100952

[2] SAR Shah e I. Biju, "Performance Comparison of Intrusion Detection Systems and Application of Machine Learning to Snort System", *Future Generation Computer Systems*, vol. 80, Elsevier BV, Mar. 2018, pp. 157-70, DOI: https://doi.org/10.1016/j.future.2017.10.016

[3] Q. Hu, Y. Se-Young y MR. Asghar, "Analysing Performance Issues of Open-source Intrusion Detection Systems in High-speed Networks", *Journal of Information Security and Applications*, vol. 51, Elsevier BV, Apr. 2020, p. 102426, DOI: https://doi.org/10.1016/j.jisa.2019.102426

[4] A. Waleed, AF. Jamali y A. Masood. "Which Open-source IDS? Snort, Suricata or Zeek", *Computer Networks*, vol. 213, Elsevier BV, Aug. 2022, p. 109116, DOI: https://doi.org/10.1016/j.comnet.2022.109116

[5] Y. Otoum, D. Liu y A. Nayak, "DL-IDS: A Deep Learning-based Intrusion Detection Framework for Securing IoT", *Transactions on Emerging Telecommunications Technologies*, vol. 33, n.° 3, Wiley, Nov. 2019, DOI: https://doi.org/10.1002/ett.3803

[6] N. Chaabouni, *et al.*, "Network Intrusion Detection for IoT Security Based on Learning Techniques". *IEEE Communications Surveys & Tutorials*, vol. 21, n.° 3, Institute of Electrical and Electronics Engineers (IEEE), 2019, pp. 2671-701, DOI: https://doi.org/10.1109/comst.2019.2896380

[7] A. P. Patil, *et al.*, "JARVIS: An Intelligent Network Intrusion Detection and Prevention System", 2022 IEEE Fourth International Conference on Advances in Electronics, Computers and Communications (Icaecc), IEEE, Jan. 2022, DOI: https://doi.org/10.1109/icaecc54045.2022.9716622

[8] Snort Project, "Snort Users Manual 2.9.16", 2020. [Internet]. Disponible en: https://snort-org-site.s3.amazonaws.com/production/document_files/files/000/000/249/original/snort_manual.pdf [Accedido: 19-oct-2022].

[9] J. F. Cañola García, "Sistema preventivo contra ataques de denegación de servicio web utilizando Deep Learning", tesis de maestría, Inst. Tecnol. Metro., Medellín, 2020.

[10] H. Asad y G. Ilir. "Diversity in Open-Source Intrusion Detection Systems". *Developments in Language Theory*, Cham, Springer International Publishing, 2018, pp. 267-81, DOI: https://doi.org/10.1007/978-3-319-99130-6_18

[11] I. Sharafaldin, Iman, A. Habibi Lashkari y A. A. Ghorbani, "Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization", Proceedings of the 4th International Conference on Information Systems Security and Privacy, Scitepress-Science and Technology Publications, 2018, DOI: https://doi.org/10.5220/0006639801080116

[12] T. Zitta, M. Neruda and L. Vojtech, "The Security of RFID Readers With IDS/IPS Solution Using Raspberry Pi", 2017, 18th International Carpathian Control Conference (ICCC), IEEE, May 2017, DOI: https://doi.org/10.1109/carpathiancc.2017.7970418

[13] S. A. Raza e I. Biju, "Performance Comparison of Intrusion Detection Systems and Application of Machine Learning to Snort System", *Future Generation Computer Systems*, vol. 80, Elsevier BV, Mar. 2018, pp. 157-70, DOI: https://doi.org/10.1016/j.future.2017.10.016

[14] A. Soucase I, "Implementación de un Sistema de Prevención de Intrusiones (IPS) en un modelo de red industrial", 2021. [Internet]. Disponible en: https://m.riunet.upv.es/handle/10251/178959 [Accedido: 19-oct-2022].

[15] K. Nam y K. Keecheon, "A Study on SDN Security Enhancement Using Open-Source IDS/IPS Suricata", 2018, International Conference on Information and Communication Technology Convergence (ICTC), IEEE, Oct. 2018, DOI: https://doi.org/10.1109/ictc.2018.8539455

[16] J. E. Cruz de la Cruz, C. A. Romero y C. Delgado, "Intrusion Detection and Prevention System for Production Supervision in Small Businesses Based on Raspberry Pi and Snort", 2020, IEEE XXVII International Conference on Electronics, Electrical Engineering and Computing (Intercon), IEEE, Sept. 2020, DOI: https://doi.org/10.1109/intercon50315.2020.9220240

[17] I. Ahmad, M. Bascheri y M. J. Iqbal, "Performance Comparison of Support Vector Machine, Random Forest, and Extreme Learning Machine for Intrusion Detection", IEEE Access, vol. 6, Institute of Electrical and Electronics Engineers (IEEE), 2018, pp. 33789-95, DOI: https://doi.org/10.1109/access.2018.2841987

[18] R. García, "Las mejores rivales y alternativas a la Raspberry Pi 4", 2022, Accedido el primero de noviembre de 2022. [Internet]. Disponible en: https://www.adslzone.net/listas/gadgets/alternativas-raspberry-pi/

[19] Elhacker.net, "Instalar y configurar IDS/IPS Suricata en una RaspBerry Pi", 2021, Accedido: primero de noviembre de 2022. [Internet]. Disponible en: https://blog.elhacker.net/2021/02/instalar-configurar-reglas-ids-ips-suricata-en-una-raspberry-pi.html

[20] Tech LBT, "Configurando Suricata en Pfsense", 2020, Accedido: 4 de octubre de 2022. [Internet]. Disponible en: https://tech.lobobrothers.com/configurando-suricata/

[21] Snort, "Rule Subscriptions Power, precision, and flexibility", 2022, Accedido: 13 de octubre de 2022. [Internet]. Disponible en: https://www.snort.org/products

[22] Tech LBT, "Implementando Pfsense con Suricata", 2020, Accedido: 4 de octubre de 2022. [Internet]. Disponible en: https://tech.lobobrothers.com/implementando-pfsense-con-suricata/