Un Modelo de Detección de Anomalías en una LAN usando K-NN y Técnicas de Computación de Alto Desempeño.

Mercedes Barrionuevo, Mariela Lopresti, Natalia Miranda, Fabiana Piccoli
LIDIC. Universidad Nacional de San Luis,
Ejército de los Andes 950 - 5700 - San Luis - Argentina
{mdbarrio, omlopres, ncmiran, mpiccoli}@unsl.edu.ar

Abstract. Detectar valores anormales a partir de grandes volúmenes de información producido por el tráfico de red ha adquirido un interés considerable en el área de seguridad de redes. Es de relevancia para todo sistema de computadoras conectadas a una red contar con un sistema de detección de eventos anómalos y un tiempo de obtención de tales eventos lo más cercano posible a su ocurrencia. Detectar valores anómalos puede conducir a los administradores de red a identificar fallas del sistema y, por lo tanto, tomar medidas preventivas antes de una masiva propagación. La detección de anomalías es un punto de partida para evitar nuevos ataques. En este artículo, presentamos una forma de pre-procesar datos para identificar anomalías mediante un algoritmo de clasificación K-NN con técnicas de computación paralelas usando Unidades de Procesamiento Gráfico.

Keywords: Tráfico de red. Anomalías. K-NN. Computación de Alto Desempeño. GPU.

1 Introducción

La variedad y complejidad del tráfico actual en Internet superan todo lo imaginado por los diseñadores pioneros de la arquitectura subyacente de Internet. Actualmente estamos inmersos en una sociedad dependiente del uso de sistemas computarizados presentes en diversas ramas como: las finanzas, industria, medicina y varios aspectos de la vida cotidiana. Con el fin de proteger o prevenir estos sistemas y la información importante o de interés para la empresa u organización, es necesario implementar tecnologías o modelos para evitar accesos no autorizados o maliciosos. Esto puede ser posible si se determinan patrones de acceso válidos.

Las amenazas a una red de datos están conformadas por un conjunto de tramas con características específicas que buscan detectar vulnerabilidades en un sistema. Estas representan riesgos, los cuales son usados para realizar ataques.

Al detectar situaciones fuera de lo común, es decir aquellas que se desvían del perfil normal de la red, los administradores se hacen preguntas tales como ¿Qué significan las desviaciones? ¿Se puede considerar tal situación como un ataque? ¿Dicha desviación pertenece a tráfico generado por nuevas aplicaciones? En base a ellos

surgen los sistemas de detección basados en anomalías, los cuales reportan toda actividad no habitual hasta ese momento, la cual puede ser normal o no.

En los sistemas cuyo objetivo es detectar ataques generalmente el margen de error suele ser alto, por ello es necesario contar con una interpretación semántica de los resultados. Esto unido a la multiplicidad de tráfico de red generado por las aplicaciones y las características tales como ancho de banda, duración de las conexiones, entre otras, hacen de esto un trabajo de alto costo computacional.

Generalmente, las investigaciones en esta área tienden a limitar la evaluación de los sistemas de detección de anomalías al cálculo de la desviación de las nuevas instancias respecto al perfil normal, constituyendo un reto convertir los resultados en reportes semánticos para los administradores de redes. En base a esto se propone un Sistema Paralelo-Supervisado de Detección de Anomalías de Red, P-SADS.

En [1] se presentó un primer modelo no supervisado de P-SADS para detectar anomalías a través de comparación de imágenes por SIFT (Scale Invariant Feature Transform) [5]. En este trabajo, se propone combinar técnicas de clasificación de tráfico y computación de alto desempeño para obtener en el menor tiempo posible buenos resultados trabajando sobre grandes volúmenes de datos. Aquí nos enfocamos en la etapa de pre-procesamiento de los datos obtenidos de la red. Se propone usar el algoritmo paralelo de clasificación supervisada K-NN *(K* vecinos más cercanos) para obtener un conjunto más pequeño de datos a procesar en la segunda etapa.

Este documento está organizado como sigue: la próxima sección describe los conceptos teóricos involucrados en el desarrollo. La sección 3 detalla las características de la primera etapa de P-SADS y en la sección 4 se muestran los resultados experimentales de su desarrollo. Finalmente, se detallan las conclusiones y trabajos futuros.

2 Marco Teórico

En este trabajo se involucran diferentes conceptos, entre los cuales se destacan el tráfico en redes de computadoras y sus anomalías, métodos de extracción de características de paquetes y algoritmos de clasificación supervisada. En esta sección describimos cada uno de ellos.

2.1 Tráfico de Datos en Redes de Computadoras

El tráfico de red proporciona información sobre qué viaja en la red. Los tipos de datos más comunes son los datos de *log* o registro, tales como registros del protocolo de Internet (TCP/IP), registros de eventos, datos de visitas a Internet, datos de informes del protocolo de gestión de red (SNMP), entre otros [10]. Esta información es de interés para la seguridad de la red, específicamente para la detección de eventos anómalos. La figura 1 ilustra un ejemplo de tráfico TCP/IP, donde las filas detallan el tráfico de red individual y las columnas son características específicas de cada tráfico. En el ejemplo, la primera columna es un índice de sesión de cada conexión y la segunda es la fecha en que se produjo la conexión [10].

```
1  06/24/1998 08:12:58 00:00:01 ntp/u 123 123 172.016.112.020 192.168.001.010 0 -
2  06/24/1998 08:12:58 00:00:01 ntp/u 123 123 172.016.112.020 192.168.001.010 0 -
3  06/24/1998 08:15:52 00:00:04 smtp 1024 25 172.016.114.169 195.115.218.108 0 -
4  06/24/1998 08:15:55 00:00:01 domain/u 53 53 192.168.001.010 172.016.112.020 0 -
5  06/24/1998 08:15:55 00:00:01 domain/u 53 53 192.168.001.010 172.016.112.020 0 -
6  06/24/1998 08:15:55 00:00:02 smtp 1025 25 172.016.114.169 196.227.033.189 0 -
7  06/24/1998 08:17:08 00:00:04 smtp 1026 25 172.016.113.084 195.115.218.108 0 -
8  06/24/1998 08:17:11 00:00:02 smtp 1027 25 172.016.113.084 196.227.033.189 0 -
9  06/24/1998 08:17:18 00:00:02 smtp 1028 25 172.016.112.149 195.115.218.108 0 -
10  06/24/1998 08:17:36 00:00:01 domain/u 53 53 192.168.001.010 192.168.001.020 0 -
11  06/24/1998 08:17:36 00:00:01 domain/u 53 53 192.168.001.010 192.168.001.020 0 -
12  06/24/1998 08:17:37 00:00:02 smtp 1029 25 172.016.114.169 194.027.251.021 0 -
13  06/24/1998 08:17:38 00:00:02 smtp 1048 25 172.016.114.169 194.007.248.153 0 -
14  06/24/1998 08:17:39 00:00:02 smtp 1049 25 172.016.114.169 197.182.091.233 0 -
15  06/24/1998 08:17:40 00:00:02 smtp 1051 25 172.016.114.169 195.115.218.108 0 -
16  06/24/1998 08:17:41 00:00:02 smtp 1052 25 172.016.114.169 196.227.033.189 0 -
17  06/24/1998 08:17:45 00:00:01 smtp 1104 25 172.016.114.169 135.008.060.182 0 -
19  06/24/1998 08:18:07 00:00:01 eco/i - - 192.168.001.005 192.168.001.001 0 -
20  06/24/1998 08:18:07 00:00:01 eco/i - - 192.168.001.005 192.168.001.001 0 -
```

Figura 1: Ejemplo de trafico TCP/IP

Los datos viajando por la red pueden proporcionar información importante en referencia a los comportamientos del usuario y del sistema. Estos tipos de datos se pueden recopilar con productos comerciales o programas específicos. Los datos TCP/IP pueden ser capturados mediante distintas herramientas, denominados sniffers[1].

El tráfico de red está compuesto de paquetes, flujos y sesiones. Un paquete es una unidad de datos enviada entre una fuente y un destino en Internet u otra red basada en TCP/IP; un flujo de red es una secuencia unidireccional de paquetes entre dos puntos finales; y los datos de sesión representan la comunicación entre computadoras. Una comunicación implica el intercambio de varios flujos. Tradicionalmente, un flujo IP contiene un conjunto de atributos, los de mayor interés son: dirección IP origen y destino, puerto origen y destino, y tipo de protocolo. Este último puede ser de capa 2 (TCP o UDP) o 3 (ICMP) si se tiene en cuenta el modelo de 4 capas de TCP/IP. Esta información permite establecer una línea base de comportamiento o patrón normal del tráfico de red haciendo más fácil identificar un comportamiento inesperado o no deseado, denominado tráfico anómalo. Por lo tanto, una estrategia de análisis por anomalías, se basa en la descripción del tráfico de comportamiento normal, clasificando como anomalía a todos los patrones que se alejen de él.

Para obtener el conjunto de datos antes mencionados existen diversas técnicas, algunas son mencionadas en la siguiente subsección.

2.1.1 Análisis y Extracción Específica de Datos de un Paquete

Cuando se estudian los aspectos particulares del tráfico de red, es necesario extraer sólo la información de los paquetes de datos para luego procesarlos. Existen diferentes técnicas de extracción y procesamiento, algunas de ellas son:

- *Representación gráfica de los datos en bruto:* Las representaciones son generalmente en forma de gráficos de dispersión 2D y 3D, gráficos basados en tiempo, histogramas, gráficos circulares o diagramas.

[1] Un **sniffer** es un programa para monitorear y analizar el tráfico en una red de computadoras, detectando los cuellos de botellas y problemas que existan en ella.

- *Información estadística y extracción de patrones*: Basados en cálculos de promedios, distribuciones de tiempo y funciones de distribución de probabilidad.
- *Análisis basado en reglas (firmas), detección de anomalías y políticas:* son todos los análisis de inspección de tráfico en busca de coincidencias con una determinada regla o firma. Las reglas se definen como valores de ciertos campos del encabezado o de una combinación de varios de ellos. Estas técnicas se utilizan en sistemas de detección de intrusos (IDS), como Snort.
- *Análisis basado en el flujo:* se centran en el tratamiento de tráfico de la red como flujos. Como la mayoría de la información intercambiada en una red es orientada a conexión y no orientado a paquetes, el análisis puede tomar ventaja de ello. Un claro ejemplo de flujo de red típico es una conexión TCP, donde los datos intercambiados se rigen por la máquina de estado TCP [8].

Cada una de estas técnicas es adecuada para determinadas situaciones, también es posible realizar una combinación de ellas. Este trabajo se basa en el análisis de flujos.

2.1.2 Ataques Comunes

Uno de los mayores desafíos de los administradores de red es detectar ataques a redes de computadoras. Un ataque consiste en aprovechar una vulnerabilidad de un sistema informático (sistema operativo, software de aplicación o sistema del usuario) con propósitos desconocidos y que, por lo general, causan daño. Por ello es imposible hacer una clasificación completa de todos los ataques reales y posibles debilidades de las redes cuando estas se conectan a Internet. Sin embargo, los de denegación de servicio (DoS) tienen gran interés.

Un ataque DoS proviene de una sola entidad y está destinado a hacer que los recursos o servicios de una computadora no estén disponibles para los usuarios. Existen distintos tipos, en particular este trabajo se ha enfocado en los ataques DoS *Smurf, Fraggle y Land* [3, 4], cada uno tiene las siguientes características:

- *Smurf***:** Este ataque utiliza el protocolo ICMP (Control Management Protocol) para enviar un ping de difusión con una dirección de origen falsificada. Existen diferentes formas de hacer un ping, ellas son:
  - *Ping Normal:* se envía una o más peticiones de echo ICMP a un sistema, el cual responde con una o más respuestas de echo ICMP, verificando así la operatividad del sistema remoto.
  - *Ping de Difusión***:** envía la solicitud de tipo echo ICMP a una dirección de difusión. Cada sistema responderá a quien le envió, inundándolo con respuestas de tipo echo ICMP.
  - *Ping de difusión con fuente falsificada*: se envía un ping de difusión con la dirección de origen falsificada con la dirección de la víctima. Cada sistema de la red responderá e inundará a la víctima con las respuestas. Esta operación es una combinación de los anteriores.

  El patrón considerado para reconocer este tipo de ataque consiste en analizar para el protocolo ICMP, si las direcciones IP origen y destino pertenecen a la misma red, y si la dirección destino es un mensaje de difusión.
- *Fraggle***:** Para comprobar si un sistema está funcionando se pueden utilizar herramientas basadas en UDP en vez de ICMP, inspeccionando si el sistema

está escuchando en un puerto específico o no. Esto se hace comúnmente con diferentes tipos de escaneos de vulnerabilidad usados tanto por atacantes como por administradores de seguridad. Por ejemplo, si un sistema escucha en el puerto TCP o UDP 19, cuando se establece una conexión a éste, el sistema respondería con un flujo constante de caracteres. Normalmente, el sistema origen utilizará el puerto TCP o UDP 7. Cuando el sistema origen comienza a recibir los caracteres, sabe que el sistema de destino está operativo y cierra la conexión. En un ataque *fraggle*, se envía un paquete de difusión con la dirección falsificada al puerto 19 de la víctima, si tienen el puerto 19 abierto responderá un flujo constante de caracteres a la víctima.
El patrón es semejante al de *Smurf pero* para el protocolo es UDP.

- *Land*: es un ataque utilizando el protocolo TCP. Consiste en crear un "bucle infinito", provocado por el envío de una solicitud SYN con la misma dirección IP de origen y destino. Esto hace que la computadora iterativamente se responda a sí misma logrando el bloqueo del equipo y no aceptando nuevas solicitudes. Como, además, agota todos los recursos del procesador, produce finalmente una denegación de servicio. El patrón considerado para reconocer este tipo de ataque consiste en analizar la coincidencia entre las direcciones IP origen y destino como así también la igualdad de los puertos.

Como se mencionó anteriormente existen muchos otros ataques de denegación de servicios pero la detección de patrones en ellos requiere un análisis más profundo y detallado, lo cual escapa a este trabajo.

2.2 Algoritmo de Clasificación Supervisada K-NN

El proceso de clasificación construye modelos capaces de determinar la pertenencia de un objeto a una categoría sobre la base de sus características. La clasificación es supervisada si de antemano existe un conjunto de observaciones ya clasificadas, y se conoce a cual pertenece cada observación. Los algoritmos dedicados al problema de la clasificación supervisada operan usualmente sobre la información suministrada por un conjunto de muestras, patrones, ejemplos o prototipos de entrenamiento que son asumidos como representantes de las clases [9].

En particular en este trabajo se utiliza el algoritmo de clasificación supervisada basada en criterios de vecindad llamado vecino más cercano (K-NN). El método del vecino más cercano y sus variantes está basado en la idea intuitiva de que objetos similares pertenecen a la misma clase, la clase a la que pertenece un objeto puede ser inferida a partir de la clase a la que pertenecen los objetos de la muestra de aprendizaje más parecidos. La idea de similitud es reflejada formalmente en el concepto de distancia, normalmente se utiliza la distancia euclidiana [6]. Si bien el cálculo del vecino más cercano puede resolverse usando técnicas paralelas, en [6] se ha demostrado que la implementación en paralelo usando GPU [7], obtiene muy buenos tiempos de respuesta.

3 Sistema Paralelo-Supervisado de Detección de Anomalías en una Red: P-SADS

La tarea de detectar posibles paquetes de datos anómalos en una red de computadoras es muy costosa, combinar técnicas de clasificación y Computación de Alto Desempeño (HPC) en su solución parece una buena alternativa. P-SADS es un sistema donde se combina HPC, clasificación supervisada y procesamiento de imágenes para detectar posibles ataques. La arquitectura del sistema planteado distingue dos etapas. En la primera se encuentran todas los pasos necesarios para la captura y pre-clasificación de tráfico, mientras que la segunda es la encargada de detectar posibles ataques en base a la comparación de imágenes generadas con los datos obtenidos en la etapa anterior. En este trabajo nos enfocamos en la primera etapa del sistema, la segunda fue presentada en [1].

La etapa 1 se encarga de capturar el tráfico de la red, pre-procesarlo y organizarlo en flujos de datos determinando si es normal o puede ser anómalo. En la figura 2 se muestra una representación gráfica de P-SADS.

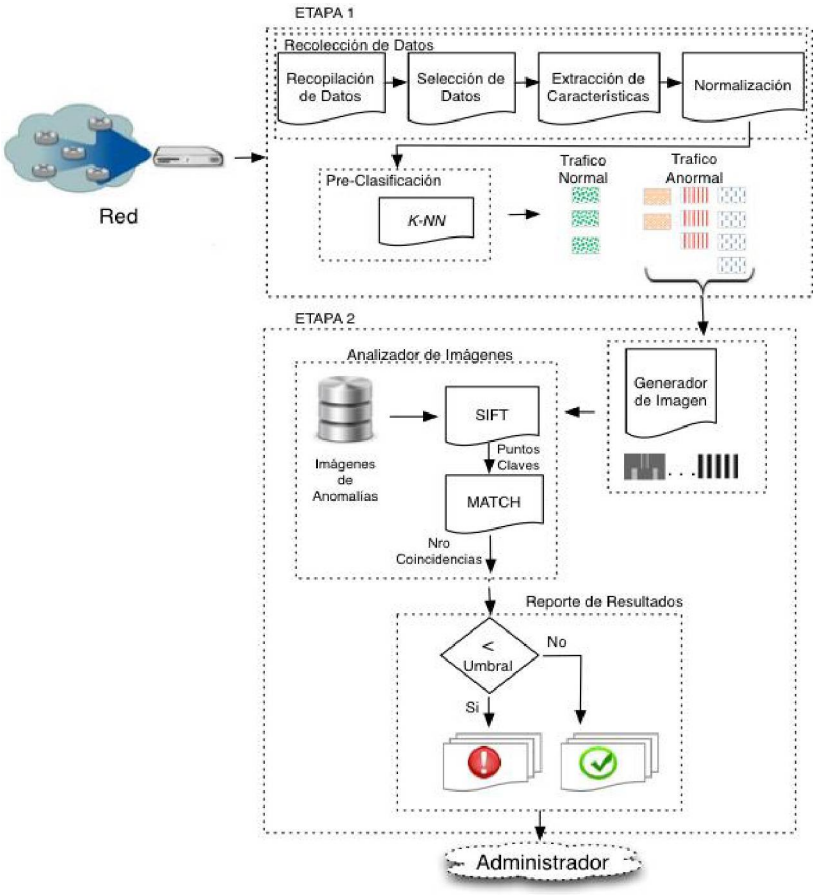

Figura 2. Esquema del modelo propuesto

Las tareas son agrupadas en la *Recolección de Datos* y su *Pre-Clasificación*, cada una con las siguientes características y funciones:

- *Recolección de Datos:* El objetivo es obtener los datos con los cuales se va trabajar, clasificarlos y reducir su volumen para la siguiente etapa. Para lograrlo se realizan en esta tarea los siguientes pasos:
  1. *Recopilación de datos*: Es el proceso de captura de tráfico de red con un sniffer y en momentos específicos donde se asume mayor tráfico de red.
  2. *Selección de datos:* En este paso se seleccionan las tramas a analizar, según los ataques considerados, son tramas TCP, UDP e ICMP.
  3. *Extracción de características:* Extrae de cada trama los campos de interés a analizar, ellos son: dirección IP origen y destino, puerto origen y destino, y tipo de protocolo de comunicación (TCP, UDP o ICMP). La figura 3 muestra ejemplos de tuplas de patrones de tráfico normal (a) y tráfico anómalo (b).

$$f_{normal1} = \{\ 10.0.0.1,\ 212.48.72.19,\ 31215,\ 80,\ UDP\}$$
$$f_{normal2} = \{\ 10.0.0.1,\ 13.29.10.199,\ 2233,\ 25,\ TCP\}$$
$$f_{normal3} = \{13.29.10.199,\ 10.0.0.1,\ ,\ ,\ ICMP\}$$

(a) Patrones de tráfico normal

$$f_{Srmurf} = \{\ 10.0.0.1,\ 10.255.255.255,\ 31245,\ 80,\ \text{ICMP}\}$$
$$f_{Land} = \{\ 10.0.0.1,\ 10.0.0.1,\ 80,\ 80,\ TCP\}$$
$$f_{Fraggle} = \{\ 10.0.0.1,\ 10.255.255.255,\ 31245,\ 80,\ \text{UDP}\}$$

(b) Patrones de ataques

Figura 3: Ejemplos de Tráfico Normal y Anómalo

  4. *Normalización*: Este paso es fundamental, las tuplas se convierten en un vector de valores enteros. La dirección IP *a.b.c.d* es transformada según la ecuación (1).

$$(a \times 256^3) + (b \times 256^2) + (c \times 256^1) + (d \times 256^0) \qquad (1)$$

- **Pre-clasificación de Datos:** Las amenazas no son el único tráfico presente en la red, los paquetes generados en un ataque van acompañados de paquetes de tráfico normal. Este módulo clasifica las tramas obtenidas en el punto anterior como tráfico normal o tráfico anómalo. Para ello, utilizamos el algoritmo de clasificación K-NN, el cual fue implementado en paralelo usando GPU. Su trabajo consiste en comparar cada vector (flujo pre-procesado), con los flujos de datos registrados y representantes de anomalías. Para ello se calcula una distancia a fin de determinar si cada nuevo flujo de datos puede considerarse una anomalía o no.

En la siguiente sección se analiza el desempeño del P-SADS en relación a su efectividad en la detección de anomalías.

4 Experimentación y Análisis de Resultados

En esta sección se presentan los experimentos realizados y el análisis de los resultados de la primera etapa de P-SADS.

La captura del tráfico de datos se hace mediante la herramienta T-Shark. Se trabajó con varias muestras de aproximadamente 2000 tramas cada una (*Recopilación de Datos*). La captura se realizó en la red LAN del Laboratorio de Redes del Departamento de Informática de la Universidad Nacional de San Luis. Se simularon los tres ataques a un servidor, buscando realizar una denegación al servicio HTTP.

Una vez obtenidas las tramas normalizadas, se construyen las bases de datos tanto con tráfico normal como anómalo. El módulo *Pre-clasificación* usando K-NN, recibe como datos de entrada estas bases de datos y realiza la evaluación para diferentes valores de K = 5, 7, 10 y 12.

Para el cálculo de K-NN se usó una computadora con una GPU Tesla K20c (2496 procesadores), 4.6 GB de Memoria y 706 MHz de Frecuencia de reloj y 2600 MHz de memoria del procesador.

Una vez obtenidos los K-NN con los distintos valores, se realizó la evaluación de los resultados en base a las siguientes métricas: *Valor Predictivo Positivo* (*Precision-PPV*), la *Tasa de Verdaderos Positivos* (*Recall-TPR*) y *F-measure (F)* [2]. Todas estas medidas pueden ser calculadas en base a los siguientes parámetros:

- *Verdadero positivo* (*TP*): es el número de tramas anómalas detectadas correctamente.
- *Falso positivo* (*FP*): número de tramas normales consideradas como tramas anómalas.
- *Falso negativo* (*FN*): cantidad de tramas anómalas no detectadas.
- *Verdadero negativo* (*TN*): número de tramas normales detectadas correctamente.

A partir de estos parámetros, las métricas se definen como:

- *Valor Predictivo Positivo*: Representa la posibilidad de que una muestra etiquetada como positiva sea efectivamente un verdadero positivo, es el porcentaje de tramas detectadas anómalas. Matemáticamente se define como:

$$PPV = \frac{TP}{(TP + FP)}$$

- *Tasa de Verdaderos Positivos o Sensibilidad*: Es el porcentaje de tramas anómalas detectadas correctamente entre todas las analizadas por el clasificador. Se calcula según:

$$TPR = \frac{TP}{(TP + FN)}$$

- *F-measure (F)*: es una métrica típica de recuperación de la información, se define como la media armónica entre las dos métricas explicadas anteriormente: *Precision* y *Recall*. Se utiliza el valor de medida F como una métrica del desempeño de K-NN, es decir el desempeño en la detección de anomalías. Su fórmula es:

$$F = 2 \times \frac{(PPV \times TPR)}{(PPV + TPR)}$$

Los resultados obtenidos con cada métrica se visualizan en la figura 4, para cada uno de los ataques considerados *Smurf* (a), *Land* (b) y *Fraggle* (c).

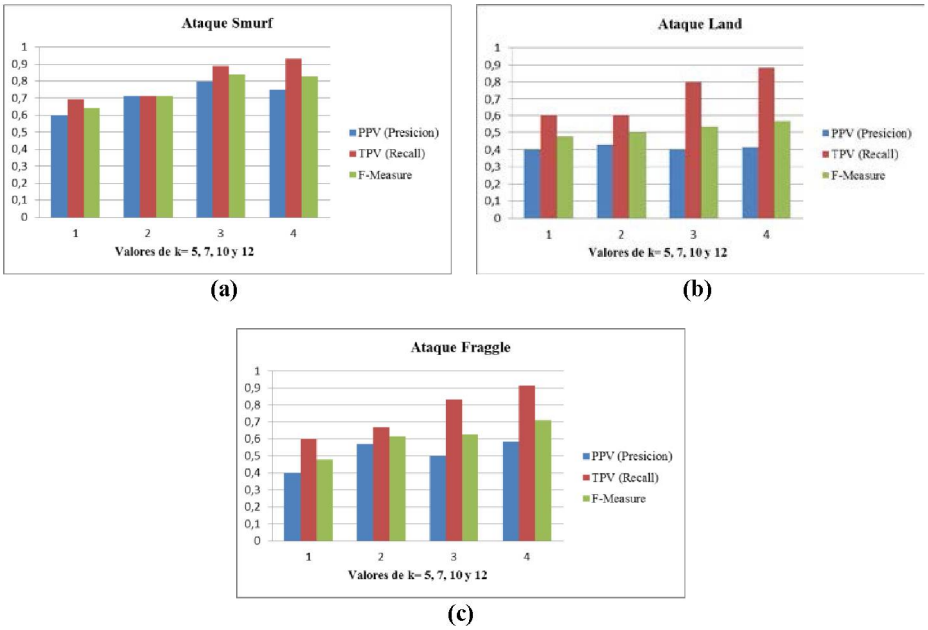

Figura 4: Resultados obtenidos en los ataques *Smurf, Land, Fraggle*.

De las gráficas anteriores se puede deducir que el ataque *Smurf* tiene una precisión aproximada del 72%, el ataque *Land* un 41% y el ataque *Fraggle* el 52%.

Con respecto a la métrica *recall* se obtuvieron resultados significativos, en promedio en *Smurf* es del 80%, en *Land* es un 72% y en *Fraggle* es un 75%. Esta métrica permitió inferir que la tasa de detección de tráfico anómalo es alta, particularmente cuando k=12.

Por último, la métrica *F-measure* también retornó resultados representativos, en particular para K=12 en los tres tipos de ataque, obteniendose un valor de 0,83 en *Smurf*; 0,56 en *Land* y 0,71 en *Fraggle*, significando que nuestro modelo tiene un buen desempeño respecto a la detección de ataques.

Por lo tanto, en base a las métricas analizadas podemos concluir que P-SADS cumple satisfactoriamente con los objetivos planteados en este trabajo.

5 Conclusiones y Trabajos Futuros

La detección de tráfico anómalo es una tarea de gran interés en la actualidad y si bien existen diversas herramientas para su detección, existe mucho trabajo por realizar

debido a la gran cantidad de datos circulantes en Internet y el constante cambio de perfil de tráfico.

Este trabajo se enfocó en la etapa de clasificación del tráfico; se propuso un modelo y se analizó su factibilidad para tres ataques conocidos del tipo de denegación de servicios: *Smurf, Land y Fraggle*. En su implementación se aplicaron técnicas HPC en GPU a fin de acelerar el proceso y obtener resultados en menos tiempo.

Se evaluó la propuesta según diferentes métricas, observándose que el modelo propuesto muestra una precisión entre el 40% y 70%, y sensibilidad entre el 60% y 83% dependiendo del tipo de ataque. Además, se evalúo el desempeño del sistema según *F-measure* obteniendo valores entre 0,5 y 0,83.

Si bien los resultados fueron satisfactorios, es necesario analizar los factores que influyen para obtener diferentes desempeños según el ataque. Otro trabajo futuro es buscar y analizar otros patrones de ataque. También, se pretende comparar nuestro desarrollo con aquellos que usan técnicas de aprendizaje de máquina o herramientas inteligentes, y en caso de ser necesario mejorar la existente.

Referencias

1. Barrionuevo, M., Lopresti, M., Miranda, N., Piccoli, M.: Un enfoque para la detección de anomalías en el tráfico de red usando imágenes y técnicas de computación de alto desempeño. XXII Congreso Argentino de Ciencias de la Computación. CACIC 2016. p. 1166-1175 (2016)
2. Davis J., Goadrich, M.: The relationship between precision-recall and roc curves, in ICML '06: Proceedings of the 23rd international conference on Machine learning. New York, NY, USA: ACM, 2006, pp. 233–240 (2006)
3. Gibson, D.: CompTIA Security+: Get Certified Get Ahead: SY0-201 Study Guide Createspace Independent Pub (2009). ISBN 9781439236369.
4. Henao Ríos J. L.: Definición De Un Modelo De Seguridad En Redes De Cómputo, Mediante El Uso De Técnicas De Inteligencia Artificial. Tesis presentada como requisito parcial para optar al título de Magíster en Ingeniería – Automatización Industrial. Universidad Nacional de Colombia. (2012)
5. Lowe, D.: Distinctive image features from scale-invariant keypoints. International journal of computer vision. Pp 91-110, (2004)
6. Miranda, N.: Cálculo en Tiempo Real de Identificadores Robustos para Objetos Multimedia Mediante una Arquitectura Paralela GPU-CPU. Tesis de Doctorado en Ciencias de la Computación. UNSL (2014).
7. Piccoli María F.: Computación de alto desempeño de GPU. 1era edic. ISBN: 9789503407592. La Plata Edulp, (2011)
8. S. Institute, Transmission Control Protocol: DARPA Internet Program Protocol Specification. Defense Advanced Research Projects Agency, Information Processing Techniques Office, (1981).
9. Tribak Hind. Análisis Estadístico de Distintas Técnicas de Inteligencia Artificial en Detección de Intrusos. Tesis Doctoral. Universidad de Granada. (2012).
10. Wang Y.: Statistical Techniques for Network Security: Modern Statistically-Based Intrusion Detection and Protection, Chapter III Network Traffic and Data, Information Science Reference - Imprint of: IGI Publishing, (2008).