Sistema de Machine Learning para la detección y prevención de ataques DDoS basados en inundación Syn

1st Cristhian Eduardo Castillo Meneses
Universidad Icesi
Cali, Colombia
cristhian.castillo1@correo.icesi.edu.co

2nd Juan Sebastian Cardona
Universidad Icesi
Cali, Colombia
juanse.bastian2012@hotmail.com

3rd Alberto Alvarez
Universidad Icesi
Cali, Colombia
alberto.alvarez512@gmail.com

4th Kevin Zarama
Universidad Icesi)
Cali, Colombia
zaramaluna1999@hotmail.com

***Resumen*—Los ataques de denegación de servicio son una manera de vulnerar los servidores para que los clientes legítimos no puedan acceder a sus servicios. Existen diferentes maneras de realizar ataques DoS y diferentes maneras de defenderse contra ellos. La inundación SYN es un tipo de ataque DoS que a través de peticiones SYN por parte de clientes falsos, consume los recursos presentes en el servidor bloqueando el acceso a clientes legítimos. Es posible, a través de algoritmos, estadísticas y comparaciones, detectar el trafico malicioso del benigno y así prevenir y detectar estos ataques. En este paper crearemos un escenario donde se realizara un ataque de inundación SYN y se detectara usando machine learning.**

***Index Terms*—DDoS, Syn Flood, Machine Lerning, Classification**

I. Introducción

Denegar un servicio, como su nombre lo indica, significa restringir o eliminar de alguna manera el acceso a él. Los Ataques DoS (Denial of Service), que se pueden dividir en DoS y DDoS (Distributed Denial of Service), buscan negar el acceso al servidor por parte de clientes reales ya que este se encuentra ocupado intentando manejar el tráfico de datos generado por los atacantes que tienen como finalidad bloquear un servidor para que no pueda comunicarse correctamente con el resto de la red y como consecuencia no pueda proporcionar sus servicios con normalidad [1]. DDoS es un tipo de DoS donde se ataca a un sistema desde múltiples orígenes con el objetivo de hacer que el sistema sea inaccesible para sus usuarios. El ataque realizado a la empresa Github el 28 de febrero de 2018 es un claro ejemplo de un ataque DDoS, GitHub quedó fuera de servicio tras un tráfico de 1.35 Tbps, enviado a través de 126.9 millones de paquetes por segundo [2]. "La disponibilidad del servicio de red o sitio web es crucial para garantizar la confianza y satisfacción del cliente, y vital para adquirir nuevos clientes en un mercado altamente competitivo", dijo Dave Larson, director de operaciones de Corero Network Security. En un estudio realizado por esta misma empresa a profesionales en seguridad informática, se obtuvo que el 45 % cree que la pérdida de confianza del cliente era la consecuencia más dañina de los ataques DDoS para sus negocios, mientras que el 34 % dijo que la perdida de ingresos era el peor defecto [3]. Según un informe realizado por NBIP junto a SIDN sobre el impacto generado por un ataque DDoS, el sector mas afectado es el gubernamental teniendo un impacto cercano a los 70M de euros [4]. Un ataque DDoS se puede realizar aprovechando diferentes vulnerabilidades, los tipos de ataque más conocidos son: ICMP, TCP SYN, UDP, TOP floods y sus combinaciones, cada uno vulnera los servidores de distinta manera [1].

Ahora bien, como existen diversas maneras de llevar a cabo un ataque DoS también existen diferentes estrategias de detectarlo y evitarlo. Es importante detectar un ataque DoS o DDoS a tiempo para evitar que los servicios que brinda nuestro servidor siempre estén activos; En el estudio realizado por Corero, empresa americana dedicada a mejorar la seguridad y disponibilidad de servicios e internet, muestra que 30 % de los encuestados confía en productos de infraestructura de seguridad tradicionales para protegerse de ataques DDoS (firewall, IPS, equilibradores de carga) pero estos pueden ser fácilmente vulnerados ya que estos no son suficientes [3]. La estrategia más prometedora es detectar y prevenir los ataques por parte del atacante, pero implica una complejidad drásticamente mayor debido a problemas como las fuentes de ataque distribuidas y la precisión de marcar el tráfico de nodos individuales como DoS [5].

En esta investigación, se hablará sobre estos tipos de ataque y sobre los métodos para blindar un servidor contra los mismos. Se creará un entorno cliente/servidor usando Python y se simulará un ataque DoS con el fin de entender la funcionalidad de estos y comprender como implementar métodos que existen para proteger un servidor contra uno de estos ataques.

II. Datasets

El proceso de recopilación de los datos para el entrenamiento del modelo de machine learning consistió en dos etapas. La primera etapa en la obtención de tráfico de red proveniente de un ataque DDoS basado en inundación Syn, que desde ahora lo denominaremos como tráfico Maliciosos, y la segunda en la generación de tráfico normal que denominaremos tráfico benigno.

II-A. Tráfico Malicioso

El dataset de tráfico malicioso se obtuvo a partir de los dataset que libera UNB[1] año tras año, en este caso se tomó el último dataset liberado en 2019 que contiene tráfico DDoS ya procesado y discriminado por tipo de ataques en archivos CSV. Este dataset es realmente útil porque contiene más de 1 millón de ejemplos con más de 80 características[2].

II-B. Tráfico Benigno

II-B1. Arquitectura:

II-B2. Captura del Trafico: Se tomó un dataset de 15000 url benignas él cual fue procesado por medio de una herramienta hecha en python que toma el dataset de URLs, verifica que aún estén activas y genera un listado con las URL activas y no activas.
Luego se tomó las URLs activas y se procede a pasarlas a una segunda herramienta que toma cada una de las URLs del listado de páginas activas y las abre en un navegador Mozilla y realiza la captura del tráfico generado por la página durante un periodo de 5 minutos. La captura del tráfico se realiza con la librería tcpdump, se genera un archivo PCAP por cada página. Posteriormente se toman los archivos PCAP guardados durante la captura del tráfico y se usa la herramienta de CICFlowMeter[3] para obtener los archivos CSV desde los PCAP generados anteriormente. Esta herramienta es la misma usada por UNB para obtener las características del dataset del dataset de tráfico maligno. Por ende se tiene las mismas características en el dataset de trafico benigno como maligno.

III. Ingeniería de Características

Se realiza una comparación entre distintos modelos de clasificación con el objetivo de verificar cual es el más apropiado para la clasificación de tráfico malicioso.

III-A. Extracción de Características

Las características fueron tomadas tomadas haciendo uso de la herramienta CICFlowMeter que extrae más de 80 características[4] correspondientes al tráfico de red.

Se asigna los valores para la variable objetivo.

Label

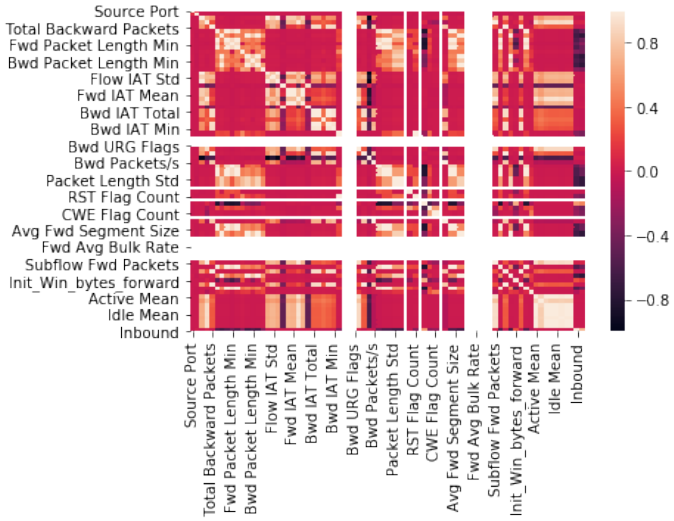
0 - Tráfico Maligno
1 - Tráfico Benigno

III-B. Análisis y preprocesamiento de los datos

El análisis exploratorio es una análisis de los datos que permite darse una idea de los datos, su estructura y relaciones utilizando en su mayoría métodos visuales. Este tipo de analisis es bastante importante porque permite identificar valores atípicos y limpiar dar a conocer si se necesita realizar una limpieza de los datos. En este caso se realiza una transformación de los datos usando el método de Yeo-Johnson con el propósito de eliminar los datos atípicos y se realiza un análisis de las variables con respecto a la variable objetivo con el fin de descartar variables que no son de utilidad para el procesamiento.

III-B1. Análisis de características: Se realiza una matriz (*Figura 1*) de correlación para poder descartar características que no serán de utilidad. Esta matriz nos permite descartar

Figura 1. Matriz de correlación de los datos para entrenamiento.
Correlación entre las variables



variables que pueden afectar en los resultados del modelo y la reducir el número de variables a ser procesadas y analizadas mejora el tiempo que tardará el sistema.
Haciendo uso del modulo feature selection de Sklearn se gráfica las características y se revisa cuales son las más importantes. En este caso entre mayor sea su valor, más relevancia tiene dentro del dataset.

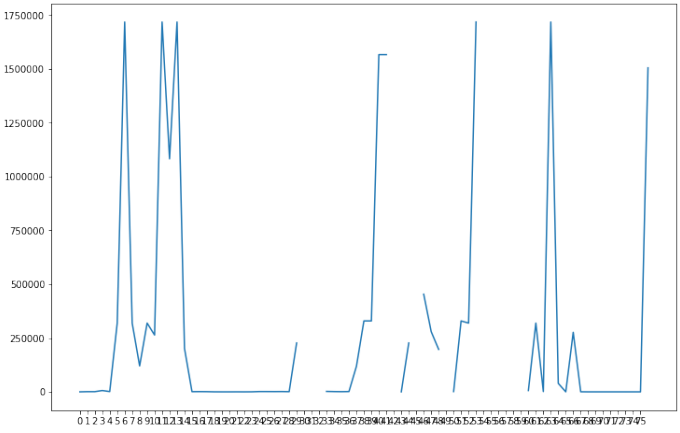
Una vez se tiene las mejores características, se descartan las características menos importantes y se entrenan los modelos.

IV. Entrenamiento y Análisis de modelos

Los modelos de entrenamiento en Machine Learning son el proceso por el cual los algoritmos de Machine Learning aprenden de los datos suministrados. En este caso se entreno

[1] https://www.unb.ca/cic/datasets/ddos-2019.html
[2] http://www.netflowmeter.ca/netflowmeter.html
[3] https://www.unb.ca/cic/research/applications.htmlCICFlowMeter
[4] http://www.netflowmeter.ca/netflowmeter.html

Figura 2. Cáracteristicas más importantes usando como parámetro el valor F de ANOVA



Algoritmo	CV Error	CV Mean
Random Forest	0.03	0.97
Gradient Boosting	0.03	0.97
AdaBoost	0.1	0.9
Multiple Layer Perceptron	0.13	0.87
Extra Trees	0.05	0.95
SVC	0.02	0.98

Cuadro I
PRESICIÓN DE LOS MODELOS CLASIFICAORES

varios modelos y se verifica la precisión predicativa de cada modelo. Luego se escoge el modelo que mejores resultados tenga y en el menor tiempo posible, se utiliza Crossvalidation para encontrar los puntajes de validación cruzada. CrossValidation divide el dataset en subconjuntos más pequeños y realiza una iteración sobre ellos en donde uno de los subconjuntos será tomado como datos test y el resto como datos de entrenamiento. La puntuación se calcula a partir del promedio obtenido en cada iteración Los algoritmos que se analizan son Logistic Regression, Support Vector Classification, K-Nearest Neighbors, Decision Tree Classifier, Random Forest Classifier, Gradient Boosting Classifier, AdaBoost Classifier, Multiple Layer Perceptron and Extra Trees Classifier como se muestra en la *tabla I*

Los mejores resultados se obtuvieron con el clasificador SVC que obtiene un índice de aciertos del 98 %. Los resultados de este modelo se presentan figura 3

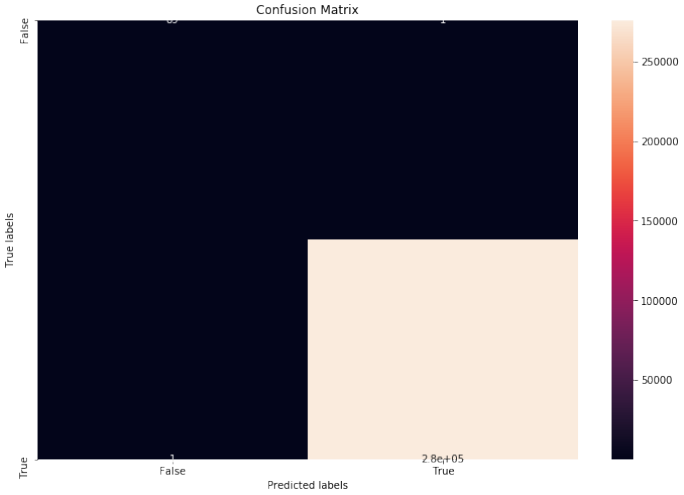
Ahora se obtuvo el modelo y las características más importantes. Es momento de probar el modelo en un entorno real.

V. IMPLEMENTACIÓN Y RESULTADOS

Se debe verificar cómo se comporta el modelo seleccionado en un ataque de tiempo real, para esto se construye una topología de red que consta de 5 host y 5 swithches creadas con mininet[5].

[5]Mininet es un emulador de red que crea redes realistas de hosts virtuales, enlaces, controladores y conmutadores

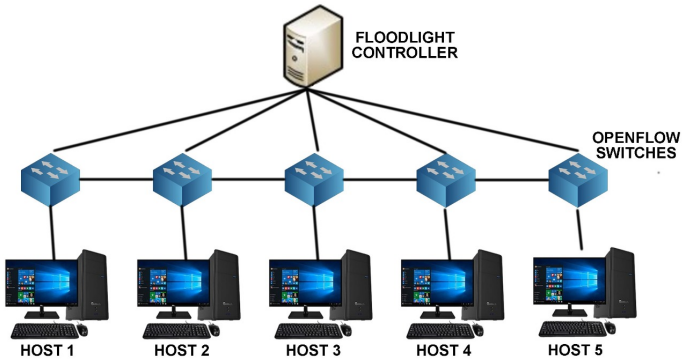
Figura 3. Confusión Matrix del modelo SVC



V-A. *Arquitectura de red*

La topología de red consta de 5 host cómo se muestra en la *Figura 4*. El host número 5 corre un servidor web http que recibe conexiones. El host 1 y 2 envían peticiones benignas al host número 1, los host 3 y 4 simulan ser bots de una botnet que realizan ataques de inundación Syn al host número 5. El ataque se realiza utilizando la herramienta Hping3. Los host en cuestión están conectados por Openflow switches[6] y controlados por un Floodlight Controller[7].

Figura 4. Topología de red para probar el sistema

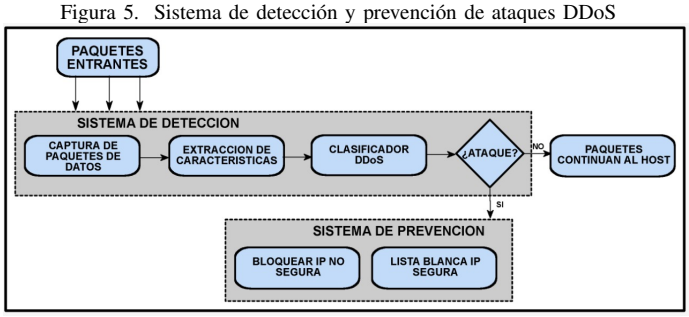


Una vez se tiene montado el entorno realista se procede a hacer el sistema que tomará capturará los paquetes que llegan a host 5, los procesa por el modelo de machine learning, quien clasifica el paquete y toma la decisión de bloquear al host que envía los ataques o no tal como se muestra en la *figura 5*

[6]Openflow Switch es un dispositivo de hardware o programa de software que reenvía paquetes en una red utilizando el protocolo de comunicaciones Openflow

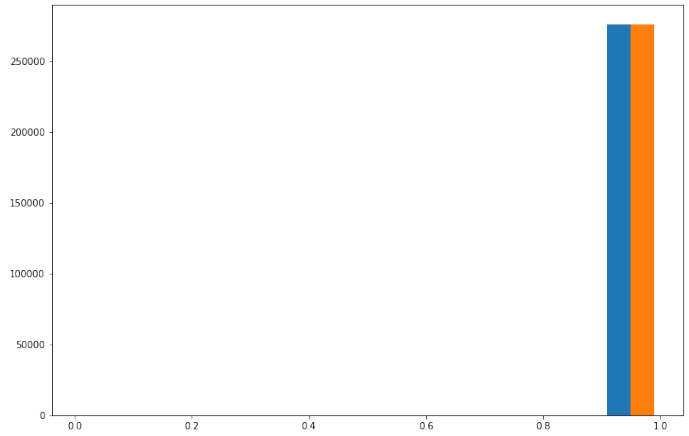
[7]El controlador de Floodlight es un controlador de red definida por software (SDN) que utiliza el protocolo de comunicaciones Openflow para controlar cómo se maneja el tráfico en un SDN.

Figura 5. Sistema de detección y prevención de ataques DDoS



El modelo captura los paquetes entrantes y los pasa a un extractor de características que extrae las características y se las pasa al modelo de SVC quien clasifica el paquete como benigno o maligno. Si el paquete es clasificado como benigno se envía al host receptor, de lo contrario se verifica una lista blanca de IPs, si la IP de origen no está en la lista blanca procede a bloquear la conexión y agregar al host emisor en una lista negra de IPs.
Este modelo fue capaz de acertar un 99 % *figura 6* de los paquetes malignos, bloqueando correctamente los hots emisores.

Figura 6. Paquetes bloqueados correctamente malignos.png



VI. Conclusiones y trabajo futuro

Los ataques de DDoS traen problemas économicos y de imagen para las organizaciones que sufren de este tipo de ataques. Hay servicios de pago que evitan este tipo de ataques de forma muy eficaz, pero que involucran cobros muy altos que algunas organizaciones pequeñas no pueden cubrir, este sistema es bastante simple y funciona en entornos pequeños ideales para organizaciones pequeñas o personas con sitios pequeños. Se propone como trabajo futuro el analizar más a fondo las carácteristicas con el próposito de encontrar el número minimo de cáracteristicas para la detección de ataques DDoS lo que podría mejorar considerablemente el rendimiento, probar el sistema en un entorno más grande con muchas más peticiones para ver si es efectivo también en entornos más grandes y caoticos; y adaptar el modelo para detectar otros tipos de ataques DDoS adicionales a inundación SYN.

Referencias

[1] G. Gonzáles. (2018) Github acaba de sobrevivir el ataque ddos más grande de la historia. [Online]. Available: HTTPS://WWW.GENBETA.COM/ACTUALIDAD/GITHUB-ACABA-DE-SOBREVIVIR-EL-ATAQUE-DDOS-MAS-GRANDE-DE-LA-HISTORIA

[2] R. S.Karthik. (2008) Analyzing interaction between denial of services (dos) attacks and threats.

[3] M. HUDSON. (2016) Ddos impact survey 2016. [Online]. Available: https://www.corero.com/company/newsroom/press-releases/loss-of-customer-trust-and-decreased-revenues-most-damaging-consequences-of-ddos-attacks-according-to-it-security-pros-and-network-operators/

[4] NBIP. (2016) Ddos-impact-report. [Online]. Available: HTTPS://WWW.NBIP.NL/WP-CONTENT/UPLOADS/2018/11/NBIP-SIDN-DDOS-IMPACT-REPORT.PDF

[5] V. S. Vinko Zlomislić, Krešimir Fertalj. (2014) Denial of service attacks: An overview. [Online]. Available: HTTPS://WWW.NBIP.NL/WP-CONTENT/UPLOADS/2018/11/NBIP-SIDN-DDOS-IMPACT-REPORT.PDF