Seguridad en la red mediante doble barrera de protección usando recursos de dominio público

Miguel Morandi[1], Alejandro Cuadra[1], Graciela Becci[1] Marcelo Gómez[1]

[1] Universidad Nacional de San Juan, Av Libertador San Martín Oeste 1109, San Juan, J5400ARL Argentina
morandi@unsj.edu.ar, acuadra@unsj.edu.ar, gbecci@unsj.edu.ar, mgomez@unsj.edu.ar

Resumen. La seguridad de una red de dominio público como la de la universidad, presenta desafíos para su defensa ante las amenazas que evolucionan en forma permanente hacia formas más sofisticadas. Existen en el Mercado opciones de defensa de la red altamente eficientes, no obstante, para una institución pública es difícil mantener actualizada una infraestructura propietaria, debido a los altos costos del equipamiento y la dependencia del fabricante para las actualizaciones del firmware y software propietario. En el presente artículo se explora la implementación de una doble barrera de defensa compuesta por un sistema de prevención de intrusión en el Router de borde y un sistema de detección de intrusión instalado aguas debajo de la red. La implementación se basa en software de dominio público ejecutado en una plataforma de hardware estándar, lo que flexibiliza la configuración y actualización de la protección mediante scripts con expresiones regulares. Se obtuvo una significativa reducción de las amenazas, quedando solamente una decena por día que son detectadas por el IDS para su posterior evaluación. Se muestra un caso de uso donde la flexibilidad de los sistemas de dominio público da la posibilidad de mantener actualizada la infraestructura, y mediante scripting poder desarrollar una alternativa eficaz para la protección a la red.

Palabras clave: seguridad en redes de datos, software de ruteo, firewall, VyOS.

1 Introducción

La red de datos de la UNSJ y el consiguiente tráfico están a cargo del IDECOM, quien administra y mantiene la red en los distintos establecimientos distribuidos geográficamente en la provincia de San Juan. La red está compuesta por un parque tecnológico heterogéneo multimarca, el cual brinda servicio de acceso a internet, campus virtual, correo electrónico, páginas web, recursos locales de almacenamiento, entre otros. Históricamente el IDECOM contaba con dos proveedores de servicio de acceso a Internet, RIU (TASA) y CLARO. En el año 2009 se gestionaron, ante LACNIC, recursos propios de Internet (ASN y Direcciones IPv4 e IPv6). A partir del año 2017 se canceló el servicio con CLARO y se contrató el servicio de ArSAT, sumados a la instalación de un nodo para el tráfico nacional a través de CABASE, alojando el IDECOM al primer nodo IXP (Internet eXchange Point) de San Juan, logrando de esta manera que proveedores locales y la UNSJ intercambien tráfico nacional. Con respecto al hardware, a esa fecha se contaba con un Router Cisco 2800, haciendo falta una actualización de equipamiento para afrontar el nuevo tráfico de la red, ya que la conexión a Internet paso de 100 Mbps a 800 Mbps, sumando todos los proveedores, incluido el tráfico en tiempo real o streaming de las videoconferencias.

Se abordaron diferentes soluciones para actualizar la administración de la red de la Universidad, entre ellas se siguió la estrategia que se describe en este artículo, consistente en actualización tecnológica en el marco de restricciones impuestas por el presupuesto de una Universidad pública nacional. Por lo que la consigna fue uso de hardware de bajo costo, actualización de hardware existente, y uso de software libre para el desarrollo de un firewall que se actualice automáticamente en forma periódica. Este upgrade era absolutamente necesario dado el nuevo incremento de tráfico de la red. La actualización tecnológica consistió en el uso de un servidor estándar dotado de mayor capacidad de procesamiento, placa de red, memoria RAM, administrado por software de dominio público como el sistema operativo de red VyOS. Actualmente se logran filtrar direcciones IPv4, ASN y dominios; se está trabajando en la implementación de la red IPV6 y sus respectivas normativas de seguridad.

Por eso este artículo se centra en el desarrollo de una doble barrera de defensa compuesta por un sistema de prevención de intrusión IPS en el Router de borde, y un sistema de detección de intrusión IDS instalado aguas debajo de la red. La implementación se basa en software de dominio público y plataformas de hardware estándar, lo que flexibiliza la configuración y actualización de la protección mediante scripts con expresiones regulares, que en forma complementaria defienden la red del tráfico indeseado.

En la Sección 2 se citan trabajos relevantes relacionados con el desarrollo de sistemas de defensa de la red usando recursos de dominio público. En la Sección 3 se detalla la metodología empleada, en la Sección 4 se describen los mecanismos para implementar la defensa, en la sección 5 se discuten los resultados, y en la Sección 6 se presentan las conclusiones y trabajo futuro.

2 Antecedentes

La eficiencia de los sistemas de detección y prevención de intrusión a la red depende de que estos sistemas estén actualizados para actuar ante amenazas que evolucionan constantemente hacia formas más sofisticadas. Es difícil mantener actualizada una infraestructura propietaria, debido a los altos costos del equipamiento y la dependencia del fabricante para las actualizaciones del firmware y software propietario, con el consiguiente riesgo de falta de soporte por obsolescencia del producto. Una alternativa cada vez más eficiente la ofrecen los sistemas basados en distribuciones de dominio público. En el caso particular del hardware de ruteo se reemplaza eficientemente mediante la implementación de un sistema operativo abierto de red ejecutado en una plataforma de hardware convencional, comportándose con todas las funcionalidades de un equipo propietario.

La Universidad pública en particular ha presentado casos de experiencia en el desarrollo de su propia infraestructura informática basada casi en su totalidad en hardware cuya actualización sea costeable y software libre. En el caso particular de los Routers la implementación con el sistema operativo de red VyOS, basado en Vyatta, tiene popularidad por las prestaciones que ofrece, fácil implementación y posibilidad de adaptarlo a las necesidades particulares de cada institución [1].

Las prestaciones de los Routers por software se pueden ampliar mediante el desarrollo de módulos de software, como es el caso del desarrollo de un driver de red wireless, no contemplado en el sistema operativo original de VyOS, con el objeto de agregar seguridad wireless a la red [2]. Como contrapartida, los Routers por software encuentran dificultad para proveer altas velocidades de tráfico debido al procesamiento de los paquetes de datos en el kernel del sistema operativo. No obstante, al tratarse de sistemas abiertos, este tipo de problemas encuentran solución en las comunidades abiertas, tal como las propuestas para mejorar la performance del

kernel optimizando el uso de recursos como es la CPU y memoria [3], o mediante la ejecución de tareas en paralelo [4] , [5] , o buscando caminos alternativos al kernel stack [6].

Los sistemas de detección de amenazas juegan un papel importante en la protección de la red mediante la generación de alertas. No obstante el análisis de estas alertas es laborioso e ineficiente cuando es realizado por un humano, y la toma de decisiones preventivas llega muchas veces tarde cuando el ataque ya penetró la red.

El uso de sistemas libres permite la implementación y automatización de estrategias de protección mediante el desarrollo de módulos de software y scripts incluyendo expresiones regulares, adaptando el firewall para prevenir ataques a las diferentes capas del modelo OSI, como por ejemplo la capa de enlace de datos, tráfico en la red y capas superiores de aplicación y acceso a la web [7], [2].

Una de las ventajas del desarrollo de firewall mediante script es que es particular de la red donde se desarrolla, dificultando el escaneo externo, que es una típica amenaza que tienen las soluciones de renombre. Por otro lado la desventaja es la necesidad de contar con personal capacitado para el desarrollo, test y actualización de un firewall dedicado. Los scripts para adaptar firewalls ofrecen la flexibilidad de poder incluir en su estrategia métodos heurísticos y expresiones regulares, como por ejemplo para inspeccionar tráfico web [8], [9].

3 Metodología de trabajo

Para afrontar las amenazas a la red incrementadas con el aumento de tráfico de datos de la UNSJ, se propuso la estrategia de usar una doble barrera de defensa, un sistema de prevención de intrusión IPS en el Router de borde, y un sistema de detección de intrusión IDS, en un servidor aguas debajo de la red, según el esquema de la **Figura 1**.

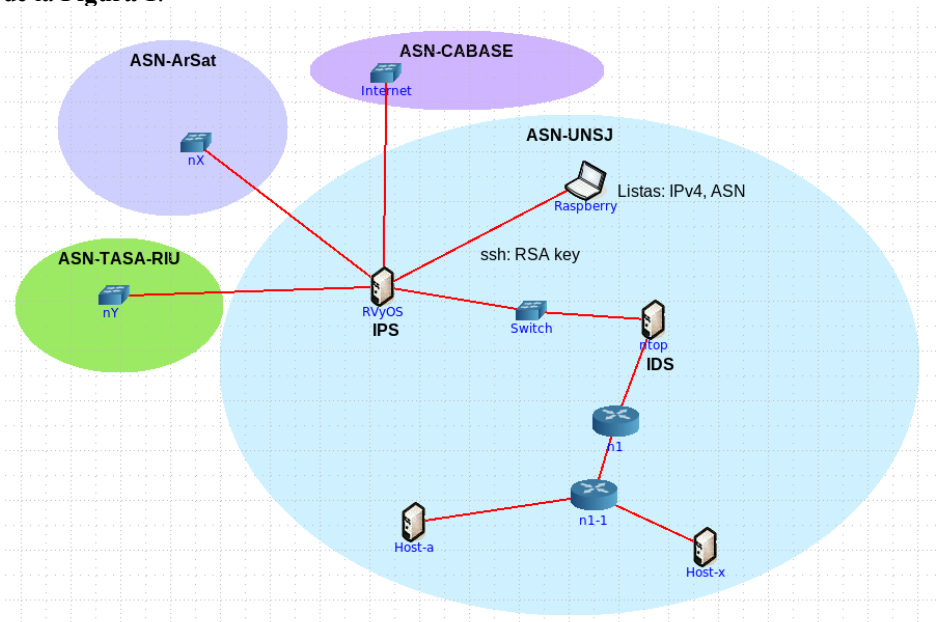

Fig. 1. Doble protección de la red: Sistema de Prevención de Intrusión **IPS**, donde el Router VyOS es el encargado de enviar el tráfico filtrado a la red interna de la Universidad, y el Sistema de Detección de Intrusión **IDS**, implementado aguas debajo de la red, complementa el filtrado del tráfico.

El Router de borde se conformó mediante un sistema operativo de red de dominio público y la actualización tecnológica con hardware de bajo costo, para el desarrollo de los firewalls descriptos a continuación.

3.1 Actualización tecnológica

A comienzos de 2018, se decidió reutilizar un servidor obsoleto para estos fines, HP Proliant DL360 de 64 bits, y convertirlo en Router. La configuración del Router significó un incremento en la capacidad de procesamiento mediante un procesador 32/64 2GHz Xeon 4 núcleos, se actualizó la placa de red a 2 Gbps y 4 Gbps (modo switch), y se incrementó la memoria RAM a 2048 MB. Con estas mejoras la plataforma quedó lista para ser convertida en Router, con la ventaja de poder realizar mejoras a futuro sin depender de licencias de marcas.

Implementación del Router por software

Con respecto al software del router se evaluó las ventajas comparativas entre implementar un sistema operativo de red y un firewall, implementados en una plataforma de hardware estándar. Por su amplia difusión se compararon pfSense y VyOS. Si bien pfSense es un firewall de red de distribución libre y de muy fácil implementación [10], en este caso para el desarrollo de un sistema de prevención de intrusión se prefirió el análisis de otro tipo de herramienta más completa para ejecutarse en un Router de borde. Por eso se eligió VyOS, que es un sistema operativo de red que soporta protocolos de ruteo BGP, OSPF y lenguajes para implementar políticas de ruteo complejas [11]. También incluye ruteo real y virtual, firewall y VPN. Con respecto a la implementación, VyOS tiene la particularidad de poder ejecutarse en una amplia variedad de plataformas físicas y virtuales, tanto en pequeñas placas x86 como en grandes servidores, en Virtual-Machines, y Kernel-based VM, en hypervisors Hyper-V y Xen. La administración de VyOS se realiza mediante interface de línea de comandos similar a los routers por hardware, y estos comandos pueden incluirse en un script para ser ejecutados en conjunto en forma automática. También permite mantener el estado de las configuraciones, archivarlas y eventualmente volver a versiones anteriores. Todas estas características fueron decisivas a la hora de invertir recursos en la curva de aprendizaje de VyOS, ganando en el manejo de una herramienta potente para la administración del tráfico de la red. Se instaló la versión VyOS 1.1.7 de 32 bits, se realizó la migración completa de la configuración del router Cisco al router VyOS.

Para agilizar el monitoreo del estado de la red de la UNSJ distribuida geográficamente en distintos departamentos, se instaló una aplicación con interface gráfica que permite el monitoreo en tiempo real de los enlaces y recursos de equipamientos críticos de la red (router, switch raíz, UPS, etc). Esta herramienta, que además tiene la capacidad de enviar alarmas por mensajes de texto, permitió encontrar cuellos de botella en varios segmentos y poder solucionarlos, logrando que en la actualidad no exista saturación de los enlaces.

3.2 Desarrollo del Sistema de Prevención de Intrusión a la red

Para prevenir el tráfico indeseado, se desarrolló un Sistema de Prevención de Intrusión a la red (**IPS**), abordando tres distintos frentes: bloqueo por hosts y redes IPv4 en listas negras, bloqueo de tráfico anunciado por sistemas autónomos en listas negras, que son identificados por el número del sistema autónomo ASN, todos con antecedentes de phishing, malware, fraude, ataques, virus, etc. Como fuente del

filtrado se buscaron otras listas además de la usada por ntop, que fueran de acceso al dominio público, dado el costo de los servicios pagos.

Pasos para Automatización del IPS

Para que el Sistema de Prevención de Intrusión IPS funcione de forma autónoma, se desarrolló un mecanismo para la automatización de la búsqueda de posibles amenazas a la red para actualización periódica del firewall, consistente en los siguientes pasos.

Paso 1: Ejecución de script de acondicionamiento de listas negras. En una placa Raspberry Pi se ejecuta el script que extrae automáticamente de las listas negras la información para clasificarla por host, network y ASN, etc., y limpia caracteres de comentarios, acondiciona los registros de la BD de spamhaus i.e.

Script en Raspberry Pi:

1. Descarga de listas negras de dominio público publicadas en Internet
2. Limpia el contenido de caracteres y comentarios para obtener las direcciones IP puras
3. Separa la información en 3 archivos usando expresiones regulares: direcciones de host, de red (IPv4), y ASN (IPv4-v6).
4. Agrega comandos VyOS

Paso 2: Envío del archivo ejecutable al Router VyOS. La placa Raspberry Pi se comunica con Router VyOS via ssh usando RSA Key y se envía el archivo.

Paso 3: Ejecución en VyOS del filtro.sh y configuración para la ejecución automática del archivo.

3.3 Implementación del Sistema de Detección de Intrusión a la red

Como Sistema de Detección de Intrusión (**IDS**), se usó ntop, con licencia para instituciones educativas, para el monitoreo del tráfico de la red. Se instaló ntopng, analizador de tráfico, y nProbe para inspección de contenido de aplicaciones en la capa 7, los cuales son capaces de analizar mediante sFlow los puertos del Router y del switch de core. Ntop recibe resúmenes de tráfico sFlow provenientes del router VyOS, que son resúmenes de paquetes de datos truncados a nivel de capa 2 de enlace, sin payload con el propósito de monitoreo de la red. Esto le permite tener un buen panorama de lo que sucede en la red. Compara este tráfico con una lista pública de hosts y networks IPv4 maliciosas, elaborada por emergingthreads.net en base hosts y redes completas comprometidas por diferentes motivos de abuso, spam, ataque, etc. Cuando ntop detecta tráfico indeseado genera una alerta. En esta implementación se generaron inicialmente entre 10000 y 15000 alertas diarias, lo cual era un considerable número de posibles amenazas.

Si bien ntop genera alertas, no bloquea el tráfico por lo que es necesario analizar las alertas para tomar acción. Para ello se guarda la información en logs y se analizan los casos particulares, esto es posible hacerlo con intervención humana dada la drástica reducción a alrededor de 10 casos diarios. No obstante, queda a futuro automatizar el proceso de actualización diaria del Router.

4 Experimentación

4.1 Implementación del IPS: bloqueo de tráfico de hosts y redes

Primero se descargaron los archivos de texto de las listas negras, para luego extraer las direcciones IPv4 de host y de redes (grupos de hosts). Para la búsqueda de direcciones IPv4 de host, se usó el comando de búsqueda grep aplicado al archivo de texto, más la expresión regular quedando el comando:

```
grep -E -o "([0-9]{1,3}[\.]){3}[0-9]{1,3}" archivo.txt
```

Si bien esta expresión regular no detecta direcciones IPv4 no válidas, como por ejemplo 256.256.256.1, se asumió que todas las direcciones del archivo de texto están conformadas y son válidas según el protocolo de direccionamiento.

Procedimiento similar se siguió para la detección de redes, cambiando solamente la expresión regular que acompaña a la búsqueda mediante el comando grep, quedando en: `"([0-9]{1,3}[\.]){3}[0-9]{1,3}\/[0-9]{1,2}"`.

Igualmente se asume que todas las direcciones IPv4 y la descripción de la red son válidas y cumplen con el protocolo de direccionamiento. Con estas direcciones de host y redes se crea un nuevo archivo que contiene solamente direcciones IP para configurar el firewall según el paso siguiente.

Tercer paso, agregar comandos VyOS. Debido a que la ejecución de scripts de migración es más simple que alterar los archivos de configuración de VyOS, es que se opta por agregar los comandos VyOS a las direcciones IP para que el script pueda ser ejecutado en el Router. Por ejemplo, se usó el comando **sed** para agregar la información necesaria para ingresar cada host en una lista de host, quedando el comando **sed**:

```
sed -i -e 's/^/set firewall group address-group malware-hosts address /'
```

Con este comando se agrega la información al comienzo de cada línea. El resto del formateo del archivo se hace mediante la ayuda de los comandos **cat** y **echo** para ejecutar los comandos VyOS. A continuación, se muestra parte del script con las acciones mencionadas:

```
cd /tmp
rm malware.sh
rm emerging-Block-IPs.txt
wget https://rules.emergingthreats.net/fwrules/emerging-Block-IPs.txt
#hosts
echo '#!/bin/vbash' >> malware.sh
echo 'source /opt/vyatta/etc/functions/script-template' >> malware.sh
echo 'configure' >>malware.sh
#echo 'del firewall name 152 rule 18' >> malware.sh
#echo 'del firewall name 152 rule 19' >> malware.sh
#echo 'del firewall name 152 rule 18' >> malware.sh
#echo 'del firewall name 152 rule 19' >> malware.sh
#echo 'del firewall group address-group malware-hosts' >> malware.sh
#echo 'del firewall group network-group malware-networks' >> malware.sh
grep -E -o "([0-9]{1,3}[\.]){3}[0-9]{1,3}" emerging-Block-IPs.txt >
malware.tmp
sed -i -e 's/^/set firewall group address-group malware-hosts address /'
malware.tmp
cat malware.tmp >> malware.sh
rm malware.tmp
#networks
grep -E -o "([0-9]{1,3}[\.]){3}[0-9]{1,3}\/[0-9]{1,2}" emerging-Block-
IPs.txt > malware.tmp
sed -i -e 's/^/set firewall group network-group malware-networks network
/' malware.tmp
cat malware.tmp >> malware.sh
rm malware.tmp
```

```
echo "commit" >> malware.sh
echo "save" >> malware.sh
echo "exit" >> malware.sh
```

El resultado es un archivo con extensión .sh listo para ser ejecutado en el router VyOS. Lo que resta por hacer es enviar el archivo al router y ejecutarlo. El archivo se envía al Router mediante el comando **scp**:

```
scp -2 -i ~/.ssh/org_rsa /tmp/malware.sh
vyos@172.16.254.1:/tmp/malware.sh
```

El archivo se ejecuta en el Router mediante el protocolo Secure Shell usando el comando **ssh**:

```
ssh -2 -i ~/.ssh/org_rsa vyos@172.16.254.1 /tmp/./malware.sh
```

Este script se ejecuta en el Router VyOS detectando las direcciones IPs. El último paso es automatizar su ejecución, para ello usamos el comando crontab para ejecutrar el script a determinadas horas.

4.2 Bloqueo de tráfico anunciado por Sistemas Autónomos ASN

El objetivo es bloquear tráfico IPv4 anunciado por Sistemas Autónomos ASN reportados en listas negras, de forma tal que el firewall se actualice automáticamente. Similarmente a lo indicado para tráfico de hosts y redes, se usaron listas negras de ASN (sistemas autónomos de BGP) de dominio público, en particular las provistas por el sitio spamhaus.org (https://www.spamhaus.org/drop/asndrop.txt)

Dicha lista se encuentra en texto plano, con comentarios (iniciados con ;) en donde en cada línea se lista un ASN, por lo que se siguió el procedimiento similar al anterior, limpiando las listas de comentarios para obtener los ASN puros. El comando **grep** más la expresión regular es:

```
grep -E -o "(AS)[0-9]{1,6}" archivo.txt
```

Esta expresión regular "(AS)[0-9]{1,6}" sólo deja pasar a los sistemas autónomos (ASN).

El próximo paso es agregar los comandos VyOS para que el script pueda ser ejecutado en el router. Para dar formato al archivo de texto puro usamos el comando sed, de manera similar a como se hizo con las IPs. El resto del formateo del archivo se hace mediante la ayuda de los comandos cat y echo. El resultado es un archivo con extensión .sh listo para ser usado en el router VyOS. Lo que resta por hacer es enviar el archivo al router y ejecutarlo. Para enviar el archivo usamos el comando **scp**:

```
scp -2 -i ~/.ssh/org_rsa /tmp/asndrop.sh vyos@172.16.254.1:/tmp/
asndrop.sh
```

El archivo se ejecuta en el Router mediante el protocolo Secure Shell usando el comando **ssh**:

```
ssh -2 -i ~/.ssh/org_rsa vyos@172.16.254.1  /tmp/./ asndrop.sh
```

El script actualiza las IPs en el router VyOS. El último paso es automatizar su ejecución mediante el comando **crontab**.

5 Resultados y discusión

Las mejoras significaron un incremento de la velocidad de la red de 2 Gbps a 4 Gbps, y se pudo implementar firewall con registro de estado (stateful) mediante IPTables a cambio de Listas de Accesso. Además, la administración de la red mediante scripting en el Shell de Linux (Bash Shell) aportó flexibilidad para implementar desarrollos de control de tráfico. Tras realizar la migración mejoró la calidad del servicio, especialmente del tráfico en tiempo real, con lo que se eliminaron los problemas en las videoconferencias, disminuyendo la latencia de respuesta a 24ms comparada con los 30 ms que tenía anteriormente. El porcentaje de uso de recursos de CPU y RAM disminuyó con la implementación de VyOS, llegando a un 10% de uso de CPU y 12% de RAM. En la Tabla 1 se resumen las mejoras introducidas por el Router VyOS.

Tabla 1. Comparación de características entre Router Cisco 2800 y Router VyOS.

	Router Cisco 2800	Router basado en servidor HP con VyOS instalado
RAM	512 MB	2048 MB
ROM	128 MB	80 GB
CPU	32 bits 250 MHz	32/64 2GHz Xeon 4 núcleos
Network	2 Gbps (2 x 1Gbps)	2 Gbps + 4 Gbps (switch)
Firewall	Firewall ACL stateful	IPtables stateful

El incremento de la tasa de transmisión se debe fundamentalmente a un upgrade en el hardware de base del Router. Por el momento no se han realizado pruebas para detectar retardos debido a procesos en el kernel [4]. En caso de detectarse latencias, y al tratarse de software libre, es factible compensar las demoras en el kernel mediante la implementación de estrategias específicas, algunas de las cuales fueron mencionadas en la Sección 2.

Los resultados de aplicar la doble barrera de protección a la red mediante un IPS y un IDS son mostrados en la Fig 2. Las capturas de pantalla de la Fig 2 a) muestra el efecto de eliminar como prueba una de las reglas del firewall. Como resultado se observaron 5278 intentos de intrusión a la red, de los cuales se detectó una dirección IP origen del mayor número de intentos.

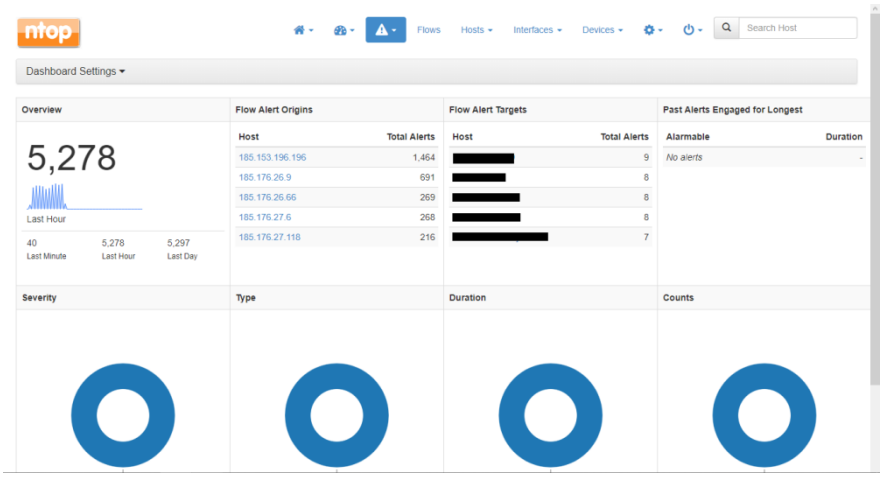

Fig 2. a) Descripción de los 5278 intentos de intrusión a la red al suspender como prueba la regla del firewall.

La Fig 2 b) da mayores detalles de las direcciones IP atacantes, tales como dirección y puerto de origen y destino, protocolo usado, etc. Para evaluar si realmente se trata de una intrusión verdadera se realizó una búsqueda de los datos de esta IP, Fig 2 c) y se comprobó que, en el caso de no ser un spoofing de dirección, es una dirección que no tiene relación con las funciones académicas de la Universidad.

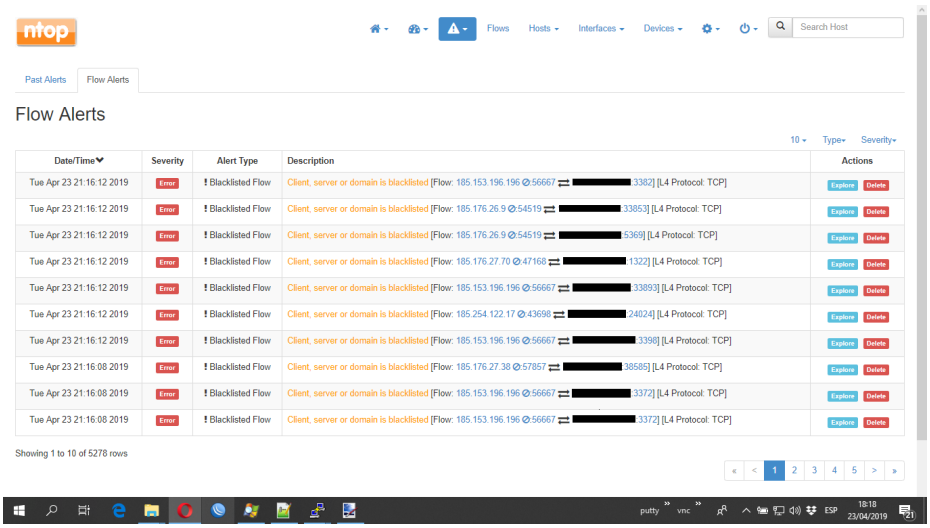

Fig 2. b) Descripción de los 5278 intentos de intrusión a la red con detalles de direcciones IP de origen, puerto y protocolo usado.

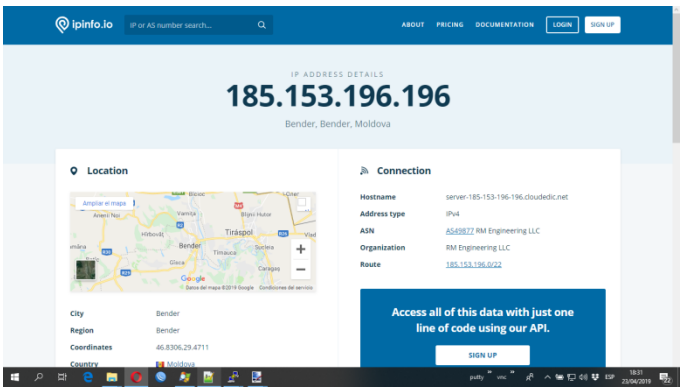

Fig 2. c) Datos (WHO_IS) de la dirección IP con mayor incidencia de intento de intrusión a la red, confirmándose que es un ataque verdadero.

6 Conclusión y trabajo futuro

La implementación de este sistema de prevención de intrusión basado en script y facilitado por el uso de un Router por software ejecutado en una plataforma genérica demuestra una vez más que los sistemas de dominio público son eficientes y flexibles para solucionar amenazas a la red, que evolucionan dinámicamente.

La estrategia de usar una doble barrera de seguridad conformada por un IPS y un IDS aguas debajo de la red, reduce significativamente el nivel de ataques a la red. No

obstante siguen existiendo casos de actividad contra la red no detectados por ambos sistemas, lo que prueba que es necesario el análisis ulterior y un automatismo del mismo para que el sistema reaccione en forma apropiada cuando se produzca una intrusión.

Como próximas tareas en relación a la performance de la red, queda determinar la incidencia del Router por software en la tasa de transmisión, y en caso de notar una ralentización, implementar algún mecanismo para sortear el cuello de botella. Con respecto a la eficacia del firewall es importante determinar si existen falsos positivos que puedan estar afectando negativamente a la red de la Universidad.

Referencias

[1] Del Brocco, A.D (last), "Gestión de Servicios Informáticos con Software libre Universidad Nacional de Quilmes," *Jorn. Argent. Softw. Libre*, pp. 30–39, 2013.

[2] S. Chaitra and R. Sharma, "Integration of Software Router with Wi-Fi for Enhanced Security," in *2017 IEEE 7th International Advance Computing Conference (IACC)*, 2017, pp. 33–36.

[3] C. Hong, K. Lee, J. Hwang, H. Park, and C. Yoo, "Kafe: Can OS Kernels Forward Packets Fast Enough for Software Routers?," *IEEEACM Trans. Netw.*, vol. 26, no. 6, pp. 2734–2747, Dec. 2018.

[4] D. Cerović, V. Del Piccolo, A. Amamou, K. Haddadou, and G. Pujolle, "Fast Packet Processing: A Survey," *IEEE Commun. Surv. Tutor.*, vol. 20, no. 4, pp. 3645–3676, Fourthquarter 2018.

[5] S. Gallenmüller, P. Emmerich, R. Schönberger, D. Raumer, and G. Carle, "Building Fast but Flexible Software Routers," in *2017 ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS)*, 2017, pp. 101–102.

[6] Z. Li, "HPSRouter: A high performance software router based on DPDK," in *2018 20th International Conference on Advanced Communication Technology (ICACT)*, 2018, pp. 503–506.

[7] M. R. Zalbina, T. W. Septian, D. Stiawan, M. Y. Idris, A. Heryanto, and R. Budiarto, "Payload recognition and detection of Cross Site Scripting attack," in *2017 2nd International Conference on Anti-Cyber Crimes (ICACC)*, 2017, pp. 172–176.

[8] I. Mukhopadhyay, K. S. Gupta, D. Sen, and P. Gupta, "Heuristic Intrusion Detection and Prevention System," in *2015 International Conference and Workshop on Computing and Communication (IEMCON)*, 2015, pp. 1–7.

[9] B. Chaudhari, P. Gothankar, A. Iyer, and D. D. Ambawade, "Wireless network security using dynamic rule generation of firewall," in *2012 International Conference on Communication, Information Computing Technology (ICCICT)*, 2012, pp. 1–4.

[10] "Getting Started With pfSense Software." [Online]. Available: https://www.pfsense.org/getting-started/. [Accessed: 20-Mar-2019].

[11] "VyOS Wiki." [Online]. Available: https://wiki.vyos.net/wiki/Main_Page. [Accessed: 20-Mar-2019].