IMPLEMENTACIÓN DE MEDIDAS DE SEGURIDAD GNU/LINUX MEDIANTE ENDIAN FIREWALL

Shaila Marquez Florez
smarquezf@unadvirtual.edu.com
Juan Manuel Reyes Mercado
jmreyesme@unadvirtual.edu.co
Fabio Rafael Ruiz Pardo
frruizp@unadvirtual.edu.co

RESUMEN: ***La seguridad en sistemas operativos GNU/Linux es un elemento fundamental para preservar la integridad, disponibilidad y confidencialidad de la información en entornos organizacionales. En esta actividad se implementan medidas de seguridad perimetral empleando la distribución Endian Firewall (EFW) dentro de un entorno virtualizado mediante VirtualBox. El desarrollo práctico permite comprender conceptos esenciales de protección de redes, tales como la segmentación por zonas (LAN, WAN y DMZ), la creación de reglas de acceso, la configuración de NAT y la aplicación de políticas de proxy con autenticación. La actividad fortalece las competencias del estudiante en la administración y aseguramiento de infraestructuras basadas en GNU/Linux, promoviendo la adopción de buenas Prácticas para la protección de servicios críticos en redes corporativas***.

PALABRAS CLAVE: GNU/Linux, Seguridad Perimetral, Endian Firewall, VirtualBox, NAT, DMZ, Proxy, Administración de Redes, Seguridad Informática, Infraestructura TI.

1 INTRODUCCIÓN

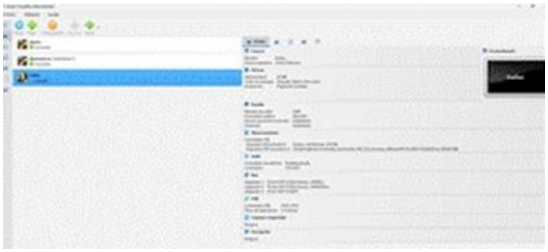
La seguridad en los sistemas operativos GNU/Linux constituye un componente esencial para garantizar la integridad, disponibilidad y confidencialidad de la información dentro de las organizaciones. En esta etapa se busca que el estudiante implemente medidas de seguridad perimetral utilizando herramientas y distribuciones especializadas como Endian Firewall (EFW), dentro de entornos virtualizados gestionados con VirtualBox. El desarrollo de la actividad permite comprender y aplicar conceptos de seguridad de redes como la segmentación por zonas (LAN, WAN y DMZ), la creación de reglas de acceso, la configuración de NAT, y la implementación de políticas de proxy con autenticación. De esta manera, se fortalece la capacidad del estudiante para administrar y asegurar infraestructuras basadas en GNU/Linux, garantizando la protección de servicios críticos y el cumplimiento de buenas prácticas de administración de sistemas.

2 CONFIGURACIÓN DE LA INSTANCIA PARA GNU/LINUX ENDIAN EN VIRTUALBOX

La instalación de Endian Firewall comienza con la descargar la imagen ISO de instalación desde https://sourceforge.net/projects/efw/, seguidamente procedemos con la creación de una máquina virtual, asignándole memoria, disco y tres adaptadores de red: red interna para la zona VERDE (LAN segura), red interna para la zona NARANJA (DMZ), y NAT para la zona ROJA para la salida a Internet. Esta configuración permite segmentar la red y aplicar políticas de seguridad desde el inicio.

Cuando copie su manuscrito a la plantilla, las páginas se numerarán automáticamente. Por favor no quite los números de página.

Figura 1: Configuración de la máquina virtual Endian



Fuente: Autoría propia

Después de crear y configurar la máquina virtual y realizamos la segmentación de la red, que consiste en configurar las zonas Green (LAN segura), Red (Internet), Orange (DMZ) y asegurarnos de que cada una tenga sus IPs, gateways y reglas de acceso bien definidas

Figura 2: Segmentación de la red

Fuente: Autoría propia

Después de, iniciamos el proceso de instalación de Endian Firewall, comenzando por la selección del idioma que se utilizará durante la instalación.

Figura 3: Inicio de instalación de Endian

Fuente: Autoría propia

Endian Firewall está completando su configuración interna: activa servicios clave, asigna interfaces a zonas (GREEN, RED, ORANGE), genera certificados, sincroniza módulos como proxy y antivirus, y valida conectividad y rutas para garantizar un funcionamiento seguro.

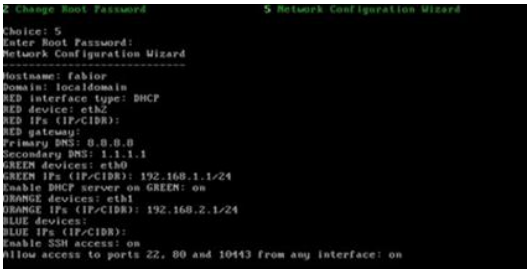
Figura 4: Configuración interna de Endian Firewall

Fuente: Autoría propia

En esta imagen se observa que Endian Firewall ha sido instalado correctamente. El sistema solicita retirar el disco de instalación y muestra la dirección IP asignada (192.168.1.1), junto con la URL de acceso a la interfaz web: https://192.168.1.1:10443

Se configuraron las interfaces de red en Endian Firewall asignando zonas e IPs: eth2 como zona RED (ROJA) con salida a Internet (DNS 8.8.8.8 y 1.1.1.1), eth0 como zona GREEN (VERDE) con IP 192.168.1.1/24, y eth1 como zona ORANGE (NARANJA) con IP 192.168.2.1/24. Además, se habilitó el acceso a los puertos 22 (SSH), 80 (HTTP) y 10443 (HTTPS).

Figura 6: Configuración de las diferentes zonas en Endian



Fuente: Autoría propia

La pantalla principal de Endian Firewall muestra que el sistema está completamente instalado y operativo, ofreciendo acceso web seguro a través de https://192.168.1.1:10443. Desde allí se puede administrar el estado del sistema, las zonas de red, los módulos activos y monitorear el tráfico y los recursos.

Figura 7: Pantalla principal de Endian totalmente configurado y funcionando

Fuente: Autoría propia

Dado que Endian Firewall no cuenta con un entorno gráfico que muestre visualmente las tres zonas (Green, Red y Orange) y su funcionamiento, accedemos a su interfaz de administración a través del navegador Firefox desde una máquina Ubuntu Desktop. Para ello, ingresamos la dirección https://192.168.1.1:10443, donde se solicita el usuario administrador y la contraseña previamente configurados en Endian.

Figura 8: Acceso a Endian Firewall desde Ubuntu Desktop

Fuente: Autoría propia

Una vez dentro de la interfaz de Endian Firewall, es posible visualizar en detalle las tres zonas configuradas: roja (Red), verde (Green) y naranja (Orange). Aunque no se presenta un diagrama gráfico como tal, la plataforma muestra de forma estructurada cómo interactúan estas zonas, incluyendo sus direcciones IP, interfaces asignadas y el flujo de tráfico entre ellas.

Figura 9: Vista general de las tres zonas configuradas

Fuente: Autoría propia

En la imagen vemos la zona roja. Esta es la parte que conecta con Internet. Endian la controla para que los equipos internos puedan salir de forma segura y para vigilar lo que entra desde afuera.

Figura 10: Funcionamiento de la zona roja

Fuente: Autoría propia

En esta ilustración vemos en funcionamiento de la zona naranja, la cual está preparada para trabajar, pero si no hay servidores conectados ni reglas que dejen pasar datos, simplemente no presenta movimiento alguno

Figura 11: Funcionamiento de la zona naranja

Fuente: Autoría propia

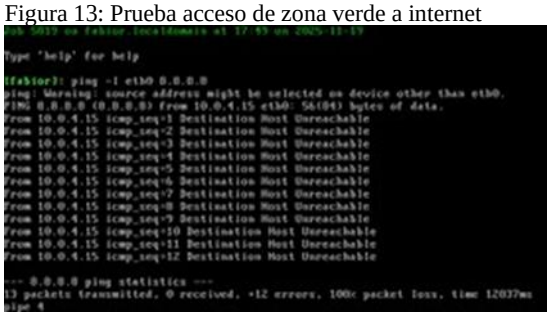
Aquí podemos ver el funcionamiento de la zona verde, esta es la red interna segura (LAN), donde se conectan los equipos de trabajo, como el Ubuntu Desktop o Server.

Figura 12: Funcionamiento de la zona verde

Fuente: Autoría propia

Prueba de salida de zona verde de Endian a internet utilizando ping -I eth0 8.8.8.8, demuestra que no hay acceso

Figura 13: Prueba acceso de zona verde a internet



Fuente: Autoría propia

Prueba de salida de la zona naranja de Endian a internet por medio del ping -I eth1 8.8.8.8 demuestra que no hay acceso a internet

Figura 14. Prueba de salida de la zona naranja a internet

Fuente: Autoría propia

Prueba de salida de la zona roja de Endian a internet por medio del ping -I eth2 8.8.8.8 demuestra que hay conexión.

Figura 15. Prueba de conexión de la zona roja a internet

Fuente: Autoría propia

3 CONFIGURACIÓN NAT.

1. Configurar la regla de NAT (Network Address Translation / Traducción de Direcciones de Red), demostrando el establecimiento de la comunicación desde la LAN hacia la WAN (Red simulada de
Internet).

Figura 16. Verificar interfaces de red

Fuente: Autoría propia

Descripción: En mi caso, observé que la interfaz enp0s3 estaba conectada a Internet mediante el adaptador NAT de VirtualBox, mientras que enp0s8 y enp0s9 estaban configuradas como redes internas, correspondientes a la LAN y la DMZ respectivamente.

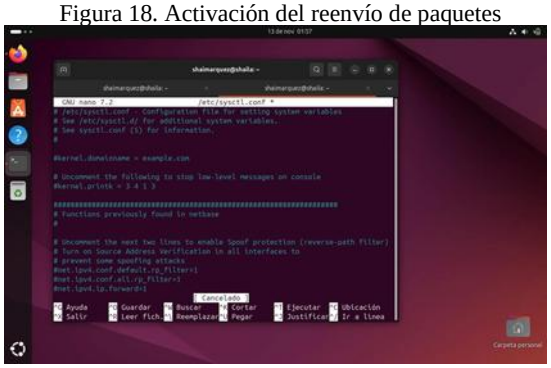
Figura 17. Asignar IP estática a las interfaces internas

Fuente: Autoría propia

Descripción: Aquí configuré direcciones IP estáticas para las interfaces internas del servidor, usando el sistema de configuración de red Netplan. Asigné a la interfaz enp0s8 la red 192.168.10.1/24, correspondiente a la LAN, y a la interfaz enp0s9 la red 192.168.20.1/24, correspondiente a la DMZ. La interfaz enp0s3 quedó con DHCP para obtener automáticamente la IP pública asignada por el adaptador NAT.

Paso 3. Activar el reenvío de paquetes Comando ejecutado: sudo sysctl -w net. ipv4.ip_forward=1

Figura 18. Activación del reenvío de paquetes



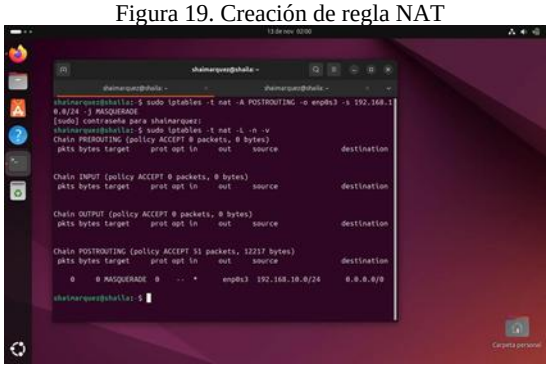
Fuente: Autoría propia

Descripción: Este paso consistió en habilitar la función de reenvío de paquetes o "IP Forwarding", que convierte a mi servidor Ubuntu en un router capaz de pasar tráfico entre redes distintas. De manera predeterminada, Linux no reenvía paquetes entre interfaces, por lo que fue necesario activar esta opción temporalmente con el comando anterior, y luego hacerla permanente editando el archivo /etc/sysctl.conf. Esta configuración es crucial, ya que sin ella los equipos de la red interna (LAN y DMZ) no podrían comunicarse con la red externa (Internet).

Paso 4. Crear la regla NAT para comunicación LAN → WAN. Descripción: En este paso creé una regla de traducción de direcciones (NAT) usando la herramienta iptables.

Esta regla indica que todo el tráfico que salga desde la red local (192.168.10.0/24) hacia Internet a través de la interfaz enp0s3 debe pasar por una máscara de dirección (MASQUERADE). Esto significa que el servidor reemplaza las IP privadas de los equipos de la LAN por su propia IP pública para que puedan navegar en Internet.

Figura 19. Creación de regla NAT



Fuente: Autoría propia

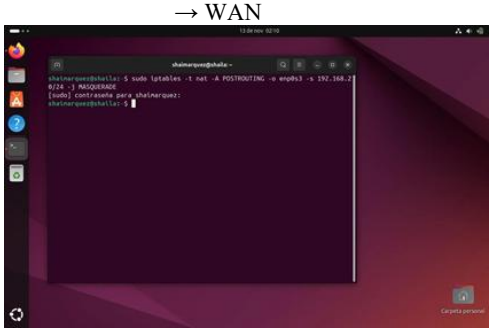
Paso 5. Verificar la conexión desde la red LAN. Descripción: En la máquina cliente de la red LAN, configuré una IP estática (192.168.10.2) y como puerta de enlace la IP del servidor (192.168.10.1). Luego utilicé el comando ping para comprobar la conectividad. Primero verifiqué la comunicación local con el servidor, después probé conectividad externa con una IP pública (8.8.8.8) y finalmente con un dominio (www.google.com). Las respuestas positivas confirmaron que el servidor estaba traduciendo correctamente las direcciones internas hacia la red externa. Este paso demostró de manera práctica el funcionamiento del NAT desde la LAN hacia Internet.

Figura 20. Verificación de la conexión

Fuente: Autoría propia

Paso 6. Crear la regla NAT para comunicación DMZ → WAN. Descripción: Aquí configuré una segunda regla de NAT, pero esta vez para la red DMZ. La lógica es la misma que la de la LAN: los equipos ubicados en la red 192.168.20.0/24 pueden salir a Internet usando la IP pública del servidor. Esto permite que los servicios alojados en la zona DMZ (por ejemplo, un servidor web o FTP) tengan acceso a la red externa de forma controlada.

Figura 21. Crear la regla NAT para comunicación DMZ → WAN



Fuente: Autoría propia

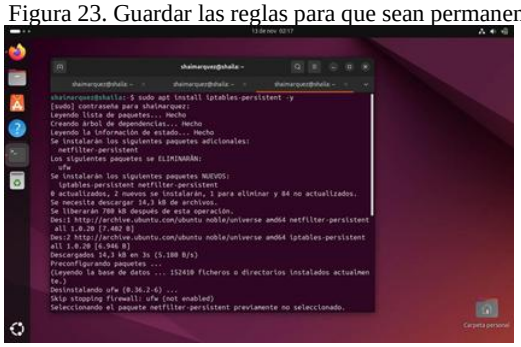
Paso 7. Verificar el reenvío de puertos y reglas NAT. Descripción: En este paso revisé las reglas activas dentro de la tabla NAT para asegurarme de que las configuraciones anteriores se aplicaron correctamente. El comando muestra en detalle todas las reglas creadas, el tráfico procesado y las interfaces involucradas. Allí pude confirmar que existían las dos reglas de tipo MASQUERADE: una para la LAN y otra para la DMZ, ambas asociadas a la interfaz de salida hacia Internet (enp0s3).

Figura 22. Verificar el reenvío de puertos y reglas NAT

Fuente: Autoría propia

Paso 8. Guardar las reglas para que sean permanentes. Descripción: Finalmente instalé el paquete iptables-persistent, que permite guardar todas las reglas configuradas para que se apliquen automáticamente cada vez que el servidor se reinicia. Sin este paso, las reglas NAT se borrarían al reiniciar el sistema, por lo que era necesario hacerlas persistentes. Con el comando sudo netfilter-persistent save confirmé que la configuración quedó guardada permanentemente. Esto garantiza que el servidor siga funcionando como router y traductor de direcciones (NAT) sin necesidad de reconfigurar todo en futuros inicios.

Figura 23. Guardar las reglas para que sean permanentes



Fuente: Autoría propia

2. Configurar la regla de NAT, demostrando el establecimiento de la comunicación de la Zona DMZ hacia la Internet. Verificar en el re-envió de puertos / NAT, la creación de las reglas.

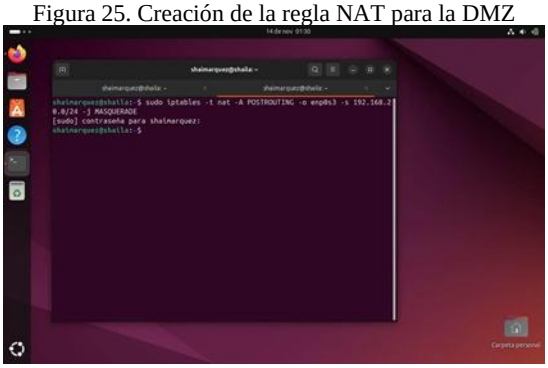
Paso 1. Verificación de la interfaz de red DMZ. Descripción: En este primer paso verifiqué las interfaces de red configuradas en el servidor Ubuntu para identificar cuál correspondía a la red DMZ. El comando ip a me permitió observar que la interfaz enp0s9 tenía asignada la dirección IP 192.168.20.1/24, la cual se configuró previamente como la red destinada para la zona desmilitarizada o DMZ.

Figura 24. Verificación de la interfaz de red DMZ

Fuente: Autoría propia

Paso 2. Creación de la regla NAT para la DMZ. Descripción: En este paso configuré una nueva regla de traducción de direcciones (NAT) con la herramienta iptables, aplicando la técnica de “enmascaramiento” o MASQUERADE. Esta regla especifica que todo el tráfico saliente desde la red DMZ (192.168.20.0/24) que se dirija a Internet debe salir a través de la interfaz externa enp0s3, reemplazando su dirección IP de origen por la dirección pública del servidor.

Figura 25. Creación de la regla NAT para la DMZ



Fuente: Autoría propia

Paso 3. Configuración de la máquina en la red DMZ. Descripción: Luego de establecer la configuración de red en la máquina cliente perteneciente a la DMZ, realicé pruebas de conectividad. Primero verifiqué la comunicación local con el servidor mediante ping 192.168.20.1, lo que confirmó que la red interna estaba activa. Posteriormente, probé la conectividad hacia Internet con ping 8.8.8.8 (servidor DNS de Google) y finalmente con ping www.google.com para comprobar la resolución de nombres.

Figura 26. Configuración de la máquina en la red DMZ

Fuente: Autoría propia

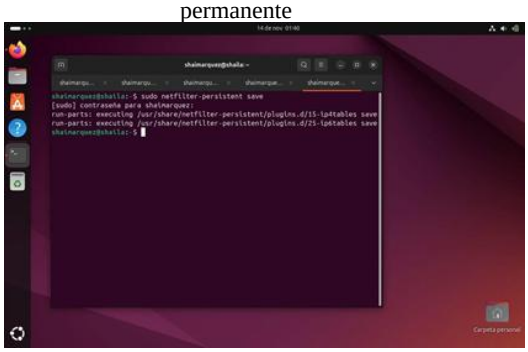
Paso 4. Verificación de las reglas NAT configuradas. Descripción: En este punto validé la existencia de las reglas de NAT previamente creadas para las redes LAN y DMZ. El comando anterior muestra en detalle todas las reglas activas dentro de la tabla NAT, junto con las interfaces involucradas, los tipos de acción aplicados y el tráfico procesado por cada una. En la salida del comando se visualizaron claramente las dos reglas MASQUERADE: una para el rango de la LAN (192.168.10.0/24) y otra para la DMZ (192.168.20.0/24), ambas utilizando la interfaz enp0s3 como salida hacia Internet

Figura 27. Verificación de las reglas NAT configuradas

Fuente: Autoría propia

Paso 5. Guardar las reglas NAT de manera permanente. Descripción: Por último, guardé la configuración de las reglas NAT para asegurar que se carguen automáticamente cada vez que el sistema se reinicie. Este paso garantiza la persistencia de la configuración de red y evita tener que volver a aplicar las reglas manualmente tras un reinicio del servidor.

Figura 28. Guardar las reglas NAT de manera permanente



Fuente: Autoría propia

4 PERMITIR SERVICIOS DE LA ZONA DMZ PARA LA RED

En esta imagen se observa el resultado del comando ip -4 addr show, donde se identifica la dirección IP asignada al servidor Ubuntu en la red DMZ. La interfaz enp0s3 muestra la IP 192.168.0.2/24, que será utilizada para las pruebas de conectividad desde la red GREEN y para la configuración de los servicios HTTP y FTP.

Figura 29. Resultado del comando ip -4 addr show

Fuente: Autoría propia

La imagen muestra la prueba de conectividad mediante el comando ping 192.168.0.2, ejecutado desde una máquina en la red GREEN. El servidor responde correctamente a las solicitudes ICMP, evidenciando que, antes de la configuración del firewall, el protocolo ICMP estaba permitido y operativo.

Figura 30. prueba de conectividad

Fuente: Autoría propia

Se presenta el comando sudo systemctl status apache2, mediante el cual se confirma que el servicio HTTP Apache se encuentra activo (running) y habilitado para iniciarse automáticamente. Esto garantiza que el servidor web está operativo dentro de la zona DMZ.

Figura 31. Comando sudo systemctl status apache2

Fuente: Autoría propia

Esta imagen muestra el proceso de instalación del paquete ProFTPD, utilizado como servidor FTP. Se evidencian los paquetes descargados, la creación de usuarios del sistema y la activación del servicio, necesarios para cumplir con el requisito de habilitar acceso FTP desde la red GREEN hacia la DMZ. Aquí se visualiza el estado del servicio FTP mediante sudo systemctl status proftpd. El servicio aparece como activo (running), indicando que el servidor FTP está correctamente configurado y disponible para recibir conexiones desde la red GREEN.

Figura 32. Proceso de instalación del paquete ProFTPD

Fuente: Autoría propia

Esta captura muestra la ejecución del comando curl http://192.168.0.2, mediante el cual se verifica que el servidor Apache2 responde correctamente a solicitudes HTTP. Esta prueba confirma que el servicio web está operativo localmente antes de evaluar accesos externos.

Figura 33. Ejecución del comando curl http://192.168.0.2

Fuente: Autoría propia

La imagen presenta la respuesta HTML completa enviada por el servidor Apache al ejecutar curl. Esto confirma la correcta entrega de la página web por parte del servidor y la disponibilidad del servicio HTTP en la red DMZ.

Figura 34. Respuesta HTML

Fuente: Autoría propia

Se observa el resultado de la conexión FTP desde un cliente en la red GREEN mediante ftp localhost o quivalente, demostrando que el servicio ProFTPD acepta conexiones correctamente y está disponible para usuarios autorizados.

Figura 35. Conexión FTP desde red GREEN

Fuente: Autoría propia

La ejecución de curl desde Endian Firewall hacia el servidor Apache2 devuelve correctamente el contenido HTML, lo que confirma que el servicio HTTP es accesible desde la red GREEN y que el NAT o enrutamiento funcionan antes del bloqueo ICMP. Endian no incluye cliente FTP, por lo que las pruebas FTP no pueden hacerse desde el propio firewall. En su lugar, se realizaron desde un equipo dentro de la red GREEN, cumpliendo el objetivo de la práctica: validar el acceso desde la red interna hacia la zona DMZ.

Figura 36. Ejecutar curl desde Endian Firewall

Fuente: Autoría propia

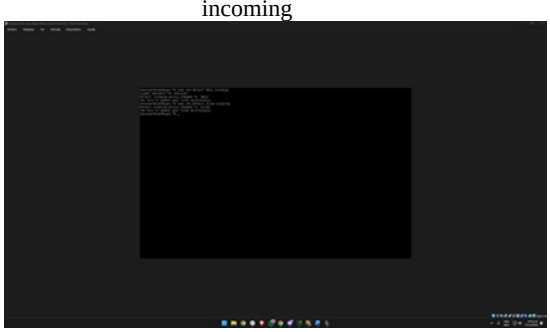
En esta captura se observa la aplicación de la política por defecto de UFW mediante sudo ufw default deny incoming. Esto asegura que cualquier conexión entrante al servidor será bloqueada, excepto las que se habiliten explícitamente (puertos 21 y 80), fortaleciendo la seguridad del servidor DMZ.

Figura 37. Aplicación de la política por defecto de UFW

Fuente: Autoría propia

La imagen muestra la ejecución del comando sudo ufw default deny incoming, donde se establece la política por defecto para denegar todo el tráfico entrante al servidor Ubuntu. Esta configuración incrementa la seguridad del sistema, permitiendo únicamente los servicios explícitamente habilitados posteriormente.

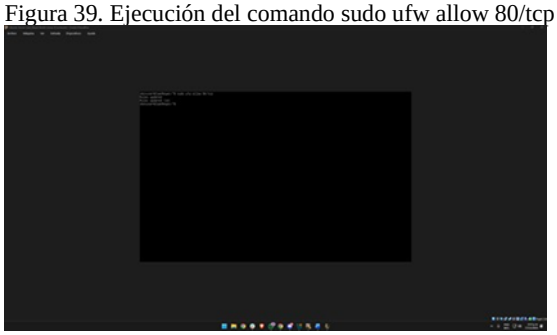
Figura 38. ejecución del comando sudo ufw default deny incoming



Fuente: Autoría propia

Se observa el comando sudo ufw allow 80/tcp, mediante el cual se permite el acceso al puerto 80 destinado al servicio HTTP. La salida confirma que la regla fue añadida exitosamente, garantizando el funcionamiento correcto del servidor web en la zona DMZ.

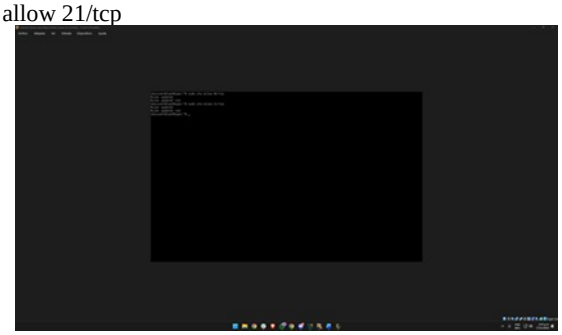
Figura 39. Ejecución del comando sudo ufw allow 80/tcp



Fuente: Autoría propia

Esta imagen muestra la apertura del puerto 21 con el comando sudo ufw allow 21/tcp. El servidor queda autorizado para recibir conexiones FTP desde la red GREEN, cumpliendo con los requerimientos de la práctica.

Figura 40. Apertura del puerto 21 con el comando sudo ufw allow 21/tcp



Fuente: Autoría propia

La captura evidencia la ejecución de un comando UFW no válido (sudo ufw enable something), lo que genera el mensaje de error “not allowed”. Este intento refleja un ajuste incorrecto que no afecta la configuración establecida, ya que las reglas válidas ya fueron aplicadas correctamente.

Figura 41. Ejecución de un comando UFW no válido

Fuente: Autoría propia

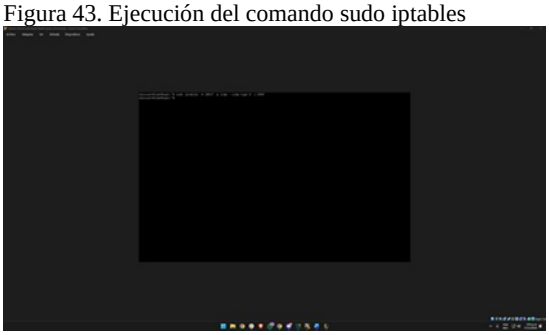
En esta imagen se presenta la salida de sudo ufw status verbose. Se puede observar que UFW está activo y que los puertos 21 (FTP) y 80 (HTTP) fueron habilitados, permitiendo conexiones desde cualquier origen. Esta figura es esencial para demostrar que los servicios requeridos están permitidos en el firewall.

Figura 42. salida de sudo ufw status verbose

Fuente: Autoría propia

Aquí se muestra el comando sudo iptables -A INPUT -p icmp --icmp-type 8 -j DROP, mediante el cual se bloquean específicamente las solicitudes ICMP de tipo echo-request (ping). Esta regla forma parte del requisito de impedir ping hacia la DMZ.

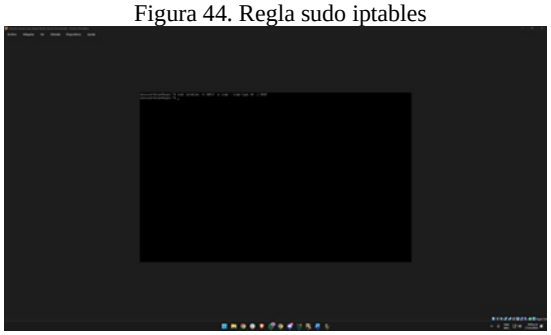
Figura 43. Ejecución del comando sudo iptables



Fuente: Autoría propia

En esta imagen se observa la regla sudo iptables -A INPUT -p icmp --icmp-type 30 -j DROP. Esta configuración bloquea el tipo ICMP 30 según lo solicitado en la guía, fortaleciendo la restricción de tráfico ICMP hacia el servidor Ubuntu.

Figura 44. Regla sudo iptables



Fuente: Autoría propia

La figura muestra la tabla completa de iptables, listando varias cadenas gestionadas por UFW. Aunque las reglas de bloqueo ICMP se insertan después, esta salida sirve para evidenciar el estado general del firewall y la estructura donde se añadieron las reglas.

Figura 45. Tabla completa de iptables

Fuente: Autoría propia

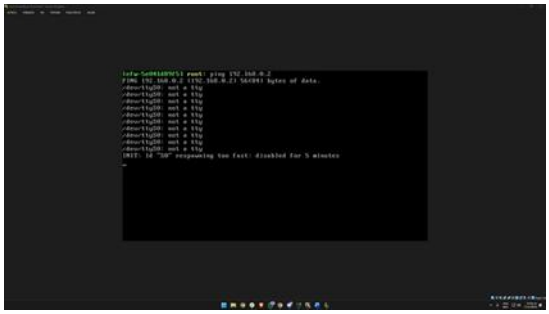
La salida del comando sudo iptables -L -v | grep icmp permite observar las reglas relacionadas con ICMP. Se destacan las líneas con DROP icmp echo-request y DROP icmp type 30, confirmando que los bloqueos fueron añadidos correctamente al firewall del servidor.

Figura 46. Salida del comando sudo iptables

Fuente: Autoría propia

En esta última imagen se observa la ejecución del comando ping 192.168.0.2 desde Endian Firewall. El servidor ya no responde a las solicitudes ICMP, validando que el bloqueo de ping es efectivo. Los mensajes del sistema (/dev/ttyS0: not a tty) no están relacionados con la prueba; lo importante es la ausencia de respuestas de ping, lo cual confirma el cumplimiento del objetivo.

Figura 47. Ejecución del comando ping 192.168.0.2 desde Endian Firewall



Fuente: Autoría propia

5 CONCLUSIONES.

La implementación de medidas de seguridad en GNU/Linux mediante la distribución Endian Firewall (EFW) permitió fortalecer las competencias en administración de sistemas operativos, demostrando la importancia de segmentar la red en zonas (LAN, WAN y DMZ) y aplicar reglas de acceso, NAT y proxy para garantizar la integridad y protección de los servicios. Este proceso evidenció cómo las herramientas de código abierto ofrecen soluciones eficientes para el control del tráfico, la autenticación de usuarios y la prevención de amenazas, consolidando así una comprensión integral de la seguridad perimetral y la gestión segura de redes bajo entornos Linux.

6 REFERENCIAS

[1] Debian. (2023). El manual del administrador de Debian 12.5.0.

[2] LPI LPIC-1 Exam 101. (2022). Tema 101: Arquitectura del Sistema. Linux Professional Institute.

[3] LPI LPIC-1 Exam 101. (2022). Tema 102: Instalación de Linux y gestión de paquetes. Linux Professional Institute.

[4] Oracle. (2020). Manual de usuario VirtualBox.