

Implementación de un Sistema de Seguridad Perimetral y Filtrado Web mediante Endian Firewall y Servidor Proxy Squid

Integrante 1 Zharith Alexandra Bedoya Yepes
zabedoyaynadvirtual.edu.co
Integrante 2 Erika Siachoque Saavedra
esiachoques@unadvirtual.edu.co
Integrante 3 Jorge Esteban Rincón Díaz
jerincon@unadvirtual.edu.co

Resumen: Este artículo técnico presenta los resultados de la implementación de un sistema de seguridad perimetral basado en la distribución especializada Endian Firewall (EFW). El proyecto se enfoca en la segmentación de redes mediante la configuración de zonas de confianza: Verde (LAN), Roja (WAN) y Naranja (DMZ), con el fin de proteger la integridad de servicios web y bases de datos críticos. A través de una metodología práctica basada en la administración por consola, se documenta la configuración de reglas de Traducción de Direcciones de Red (NAT), políticas de filtrado de tráfico para protocolos HTTP, FTP e ICMP, y la gestión de un Proxy HTTP con listas negras y autenticación de usuarios. Los resultados demuestran una arquitectura de red resiliente que cumple con los estándares de seguridad para infraestructuras GNU/Linux en entornos organizacionales.

Palabras clave— GNU/Linux, Endian, Seguridad Perimetral, DMZ, NAT, Proxy, IEEE.

I. INTRODUCCIÓN:

En la actualidad, la administración de sistemas operativos Open Source no solo demanda eficiencia operativa, sino una postura proactiva frente a las vulnerabilidades de red. La seguridad perimetral se ha convertido en el primer bastión de defensa, donde la correcta delimitación de fronteras digitales es vital para prevenir accesos no autorizados a la información sensible de las instituciones.

El presente trabajo se desarrolla en el marco de la Etapa 7 del diplomado de profundización, donde se plantea la necesidad de garantizar la protección de una intranet a través de una zona desmilitarizada (DMZ). Para ello, se utiliza la herramienta Endian UTM, la cual permite centralizar servicios de seguridad como firewalling, NAT y proxy en una sola plataforma. Durante el desarrollo de la actividad, se hace énfasis en la gestión mediante línea de comandos, asegurando que el administrador posea un control total y detallado sobre el estado de los servicios (start, stop, status) sin depender de interfaces gráficas.

A lo largo del documento, se describen de manera secuencial cinco temáticas fundamentales: desde la instalación base del firewall y la interconexión de tarjetas de red en entornos virtuales, hasta la aplicación de políticas de acceso granulares que permiten o deniegan el tráfico entre zonas según la necesidad del servicio. Esta aproximación no solo busca cumplir con un requerimiento académico, sino proporcionar una guía replicable para la implementación de soluciones de seguridad robustas bajo plataformas GNU/Linux.

II. OBJETIVOS

- Configurar un entorno de red seguro mediante zonas DMZ para proteger bases de datos y aplicaciones Web.
- Implementar y administrar un Firewall Endian para el control de tráfico entre las zonas Verde, Roja y Naranja.
- Validar la conectividad y restricciones de red mediante el uso de comandos de consola y pruebas de servicios HTTP/FTP.

III. INFRAESTRUCTURA Y RECURSOS

Para el desarrollo de la solución, se utilizó el siguiente stack tecnológico:

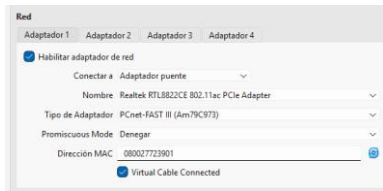
- **Hipervisor:** Oracle VirtualBox.
- **Firewall:** Endian UTM (EFW).
- **Servidor DMZ:** Ubuntu Server (Servicios Web y Base de Datos).
- **Estaciones LAN:** Clientes GNU/Linux.

IV. DESARROLLO DE TEMÁTICAS

- A. Temática 1: Configuración de la instancia para GNU/Linux Endian en Virtualbox e instalación.

Figura 1.

Configuración de red Adaptador 1.

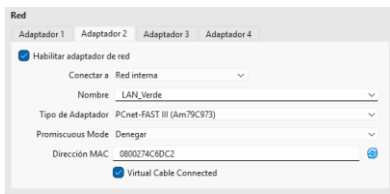


Fuente: Autoría propia

En esta imagen se muestra la sección de configuración de red de una máquina virtual en VirtualBox, específicamente para el Adaptador 1. Se observa que el adaptador está habilitado y configurado en modo Adaptador puente, utilizando la tarjeta Wi-Fi física Realtek RTL822CE de la computadora anfitriona para que la máquina virtual tenga su propia dirección IP en la red local. Además, se detallan aspectos técnicos como el tipo de adaptador PCnet-FAST III, el modo promiscuo en "Denegar", una dirección MAC específica y la confirmación de que el cable virtual está conectado.

Figura 2.

Configuración de red Adaptador 2.

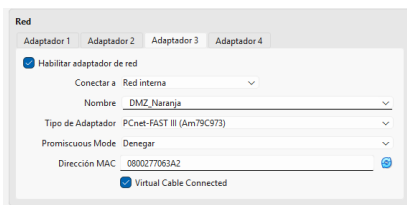


Fuente: Autoría propia

En esta imagen se muestra la configuración que he establecido para el Adaptador 2 de mi máquina virtual en VirtualBox, donde he habilitado el adaptador de red conectándolo específicamente a una Red interna denominada "LAN_Verde". He mantenido el tipo de adaptador en PCnet-FAST III con el modo promiscuo denegado, y se puede observar la dirección MAC generada para esta interfaz, asegurándome de que la opción de "Cable conectado" esté activa para permitir la comunicación interna entre las máquinas de este segmento.

Figura 2.

Configuración de red Adaptador 2



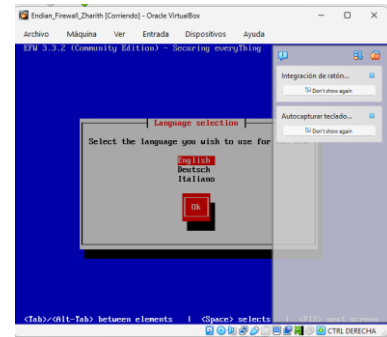
Fuente: Autoría propia

En esta captura presento la configuración de mi Adaptador 3, el cual he habilitado para funcionar bajo una

Red interna que he nombrado "DMZ_Naranja", permitiendo así un aislamiento controlado dentro de mi entorno virtual. Al igual que en las interfaces anteriores, utilizo el tipo de adaptador PCnet-FAST III con el modo promiscuo configurado en "Denegar" y una dirección MAC propia, manteniendo marcado el "Cable conectado" para asegurar que este segmento de red esté operativo y listo para la comunicación de datos.

Figura 3.

Instalación de Endian Firewall.

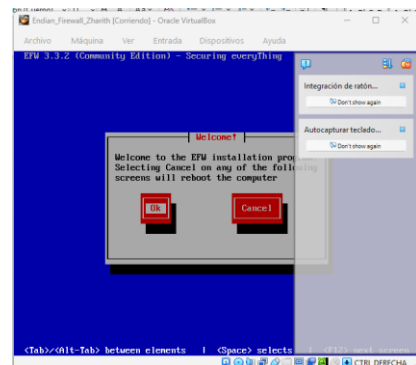


Fuente: Autoría propia

En esta captura muestro el inicio de la instalación de mi sistema Endian Firewall (EFW) 3.3.2 en Oracle VirtualBox, donde me encuentro en la pantalla de selección de idioma para el proceso de configuración. He nombrado a la máquina virtual como "Endian_Firewall_Zharith" y actualmente tengo resaltada la opción de inglés, lista para avanzar tras confirmar con el botón "Ok". Se puede apreciar que la máquina está en ejecución y el entorno me ofrece las opciones de navegación típicas mediante teclado para interactuar con la interfaz de consola del instalador.

Figura 4.

Pantalla de bienvenida de Endian Firewall.



Fuente: Autoría propia

En esta imagen muestro la pantalla de bienvenida del programa de instalación de Endian Firewall (EFW) 3.3.2 en mi máquina virtual, donde se me indica que cancelar el proceso en cualquiera de las siguientes etapas reiniciará el

equipo. Me encuentro interactuando con la interfaz de usuario en modo texto dentro de VirtualBox, teniendo seleccionado el botón "Ok" para proceder formalmente con la instalación de este sistema de seguridad. Se puede observar que sigo trabajando sobre la máquina denominada "Endian_Firewall_Zharith" y que el entorno está listo para continuar con la configuración del software.

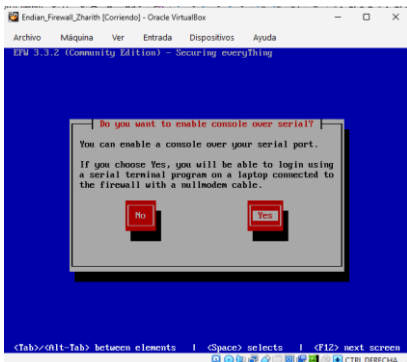
Figura 5.
Instalación de Endian Firewall.



Fuente: Autoría propia.

En esta imagen muestro el momento en que el instalador de Endian Firewall (EFW) 3.3.2 me solicita confirmación para preparar el disco duro en /dev/sda, advirtiéndome que este proceso particionará la unidad y eliminará todos los datos existentes. Me encuentro en esta etapa crucial de la configuración dentro de mi máquina virtual en VirtualBox, donde debo seleccionar la opción "Yes" para permitir que el sistema cree el sistema de archivos necesario y continúe con la instalación. Se puede observar que la interfaz sigue siendo en modo texto, manteniendo la coherencia con los pasos previos que he venido realizando en este entorno.

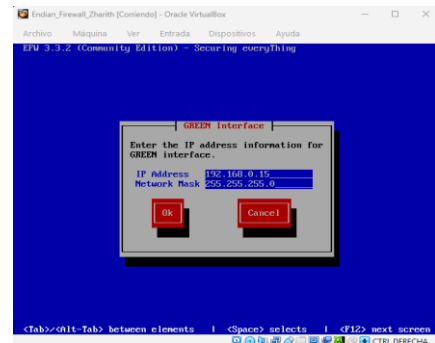
Figura 6.
Instalación de Endian Firewall.



Fuente: Autoría propia.

En esta captura de mi proceso de instalación en VirtualBox, me encuentro ante la pregunta de si deseo habilitar la consola a través del puerto serial en mi sistema Endian Firewall (EFW). El mensaje me explica que, al seleccionar "Yes", podría iniciar sesión utilizando un programa de terminal serial desde otro equipo conectado mediante un cable null-modem; sin embargo, en este punto de la configuración de mi máquina virtual "Endian_Firewall_Zharith", tengo seleccionada la opción "No" para continuar con la instalación estándar basada en la interfaz de pantalla actual. Se observa que el entorno de texto sigue siendo el medio principal para gestionar estas preferencias iniciales del sistema de seguridad.

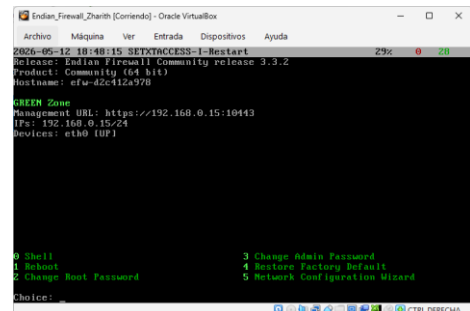
Figura 7.
Parámetros de red de Endian Firewall.



Fuente: Autoría propia.

En esta captura de mi proceso de configuración en VirtualBox, me encuentro definiendo los parámetros de red para la interfaz GREEN de mi sistema Endian Firewall, la cual actuará como mi red local segura. He asignado manualmente la dirección IP 192.168.0.15 con una máscara de red 255.255.255.0 para identificar este nodo dentro de mi topología. Actualmente tengo seleccionado el botón "Ok", listo para confirmar estos datos y avanzar con el despliegue de mi máquina virtual "Endian_Firewall_Zharith" en este entorno de consola.

Figura 8.
Parámetros de red de Endian Firewall.

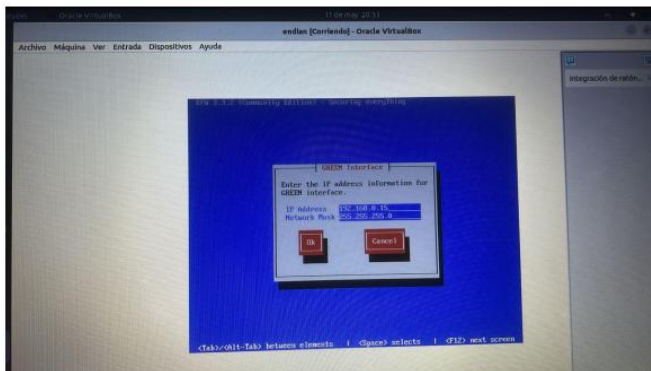


Fuente: Autoría propia.

B. Temática 4: Reglas de acceso para permitir o denegar el tráfico.

Figura 29.

Interface Ip Address-Network Mask.

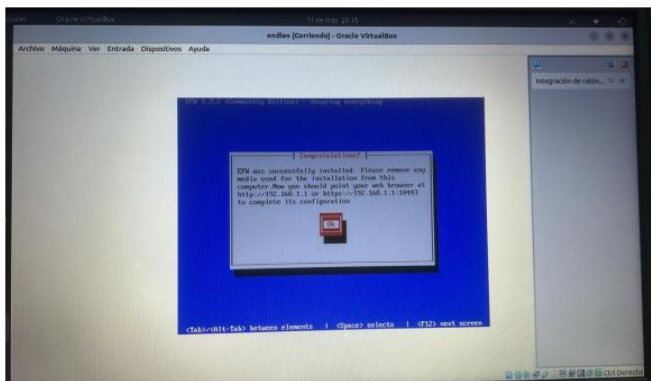


Fuente: Autoría propia.

Para la implementación del entorno de red segmentado, se hizo uso del hipervisor Oracle VirtualBox con el fin de desplegar el firewall de código abierto Endian Firewall Appliance (EFA) en su versión 3.3.2 (Community Edition). Durante la fase de inicialización del sistema operativo dentro de la máquina virtual, se procedió con la configuración manual de la interfaz GREEN (red interna/segura). Como se ilustra en la interfaz de usuario basada en texto, se asignó de forma estática la dirección IP fija 192.168.0.15 junto con una máscara de subred de 24 bits (255.255.255.0), estableciendo así el direccionamiento base para la red de área local (LAN) protegida."

Figura 30.

Efw Successfully



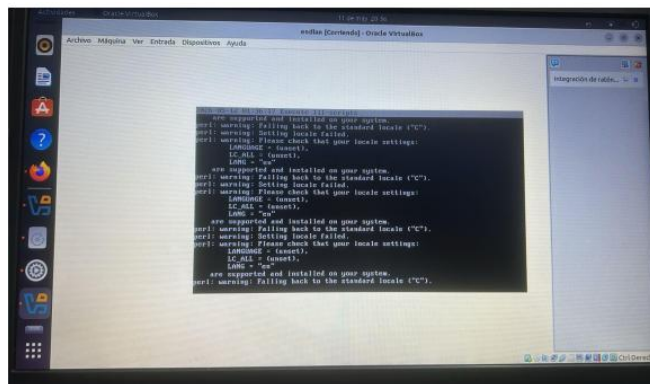
Fuente: Autoría propia.

Figura 31.

Una vez finalizado el proceso de copia de archivos y configuración inicial, el asistente en entorno de texto confirmó la instalación exitosa de Endian Firewall Appliance

(EFA). Tras recibir esta notificación, se procedió a desmontar el medio de instalación virtual en Oracle VirtualBox para evitar bucles de reinicio. Asimismo, se registro las directrices del sistema que indican el acceso a la consola de administración web mediante los protocolos HTTP y HTTPS, específicamente a través de las URLs <http://192.168.1.1> o <https://192.168.1.1:10443>, con el propósito de iniciar la fase posterior de aprovisionamiento y políticas de seguridad.

Install

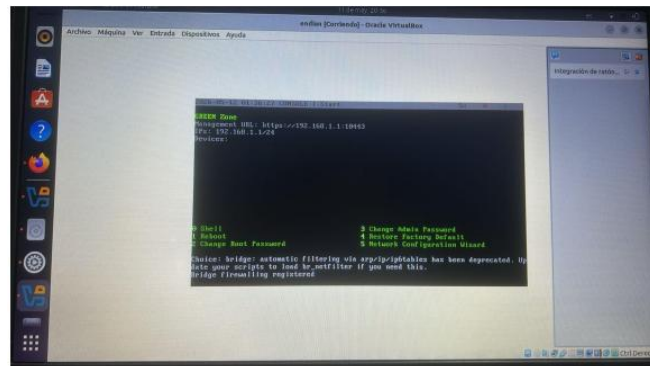


Fuente: Autoría propia.

Durante el primer arranque del sistema de seguridad, documenté la ejecución automática de los scripts de inicialización del núcleo (Execute ini-scripts). Como se evidencia en la salida de la consola de comandos de la máquina virtual, el intérprete Perl emitió una serie de advertencias estándar (perl: warning: Setting locale failed.) debido a que las variables de entorno LANGUAGE y LC_ALL se encontraban sin asignar (unset). A pesar de este comportamiento en la localización regional, el sistema aplicó de forma automática la configuración por defecto (Falling back to the standard locale "C"), permitiendo que los servicios críticos del cortafuegos continuaran su secuencia de carga sin comprometer la estabilidad ni la integridad del software

Figura 32.

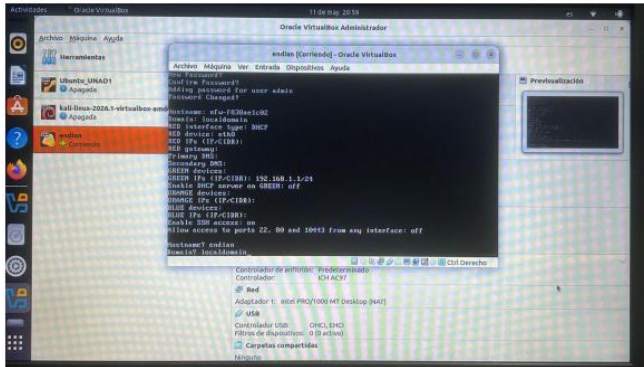
Green zone



Fuente: Autoría propia.

Figura 33.

Ajustes Hosname- Domain- Red Interface- Dns. Orange – Green

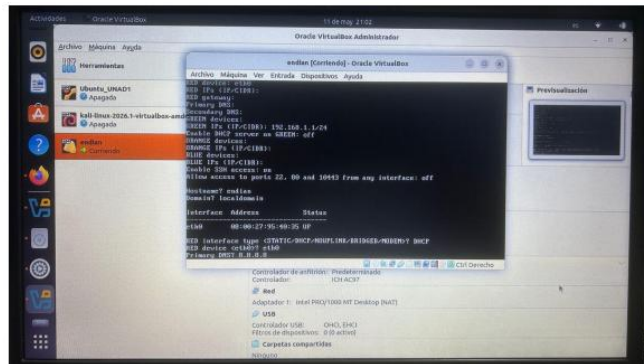


Fuente: Autoría propia.

Posterior al arranque inicial, se accedió a la consola de administración por línea de comandos para realizar el aprovisionamiento global del sistema de seguridad. En primer lugar, se efectuó el cambio de credenciales por defecto asignando una nueva contraseña robusta para el usuario administrador (Admin). Acto seguido, el sistema desplegó el mapa general de direccionamiento de red, donde se verifico los parámetros asignados a la interfaz externa RED (vía DHCP a través de eth0) y a la interfaz interna GREEN (192.168.1.1/24), manteniendo deshabilitados los segmentos ORANGE (DMZ) y BLUE (Wireless). Finalmente, se procedió con la asignación de la identidad del dispositivo en la red local, definiendo el nombre de host como Endian y el sufijo de dominio bajo la nomenclatura localdomain

Figura 34.

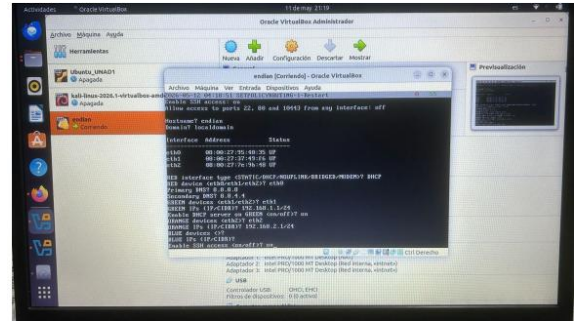
Interface Address Status



Fuente: Autoría propia.

Figura 35.

Resultado de la configuración

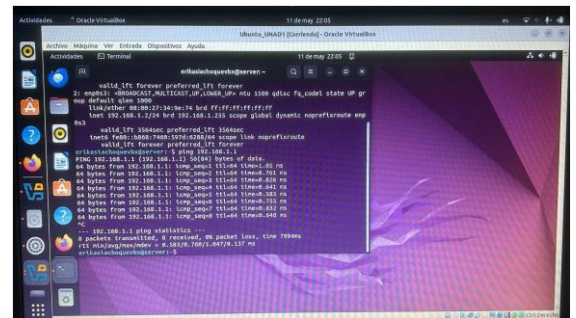


Fuente: Autoría propia.

La configuración de red del firewall Endian Firewall Community se realizó correctamente, implementando las zonas GREEN, RED y ORANGE requeridas para la segmentación de la red. Con esta configuración se estableció la base para continuar con la implementación de servicios, reglas NAT, control de acceso y pruebas de conectividad dentro del entorno virtualizado.

Figura 36.

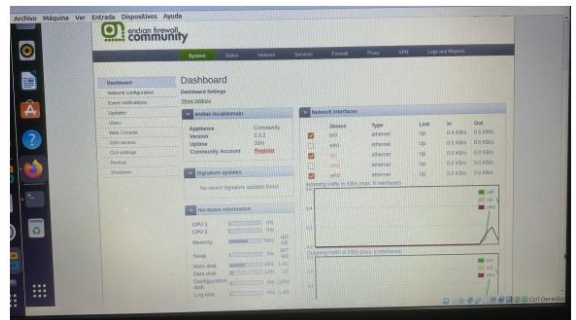
Conexión con Endian desde Ubuntu.



Fuente: Autoría propia.

Figura 37.

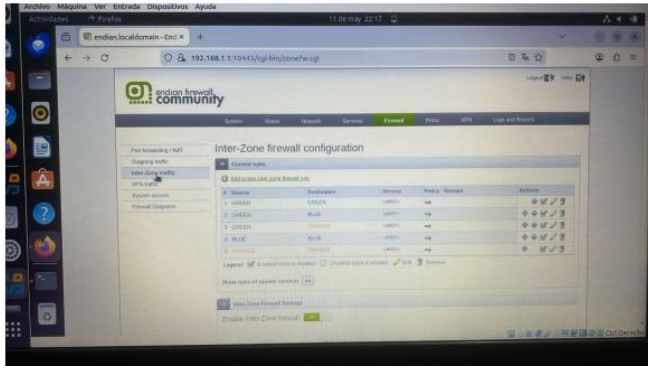
Dashboard



Fuente: Autoría propia.

Figura 38.

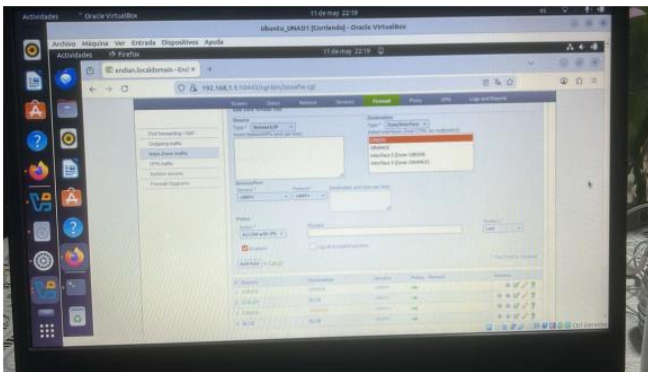
Inter-Zone Firewall Configuration



Fuente: Autoría propia.

Figura 39.

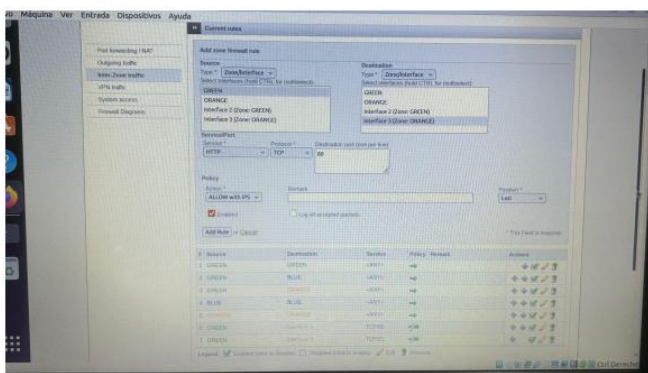
Firewall Add Rule



Fuente: Autoría propia.

Figura 40.

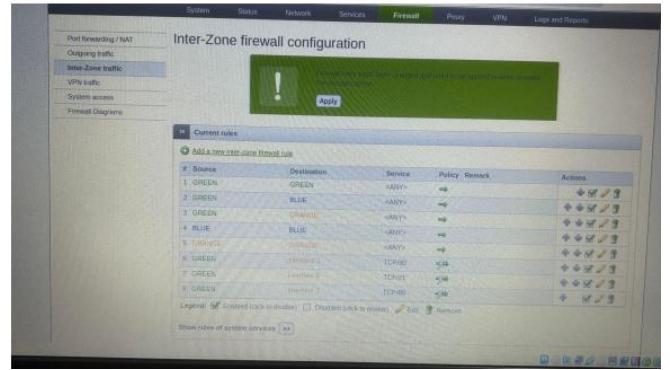
CONFIGURANDO Regla Http Regla FTP



Fuente: Autoría propia.

Figura 41.

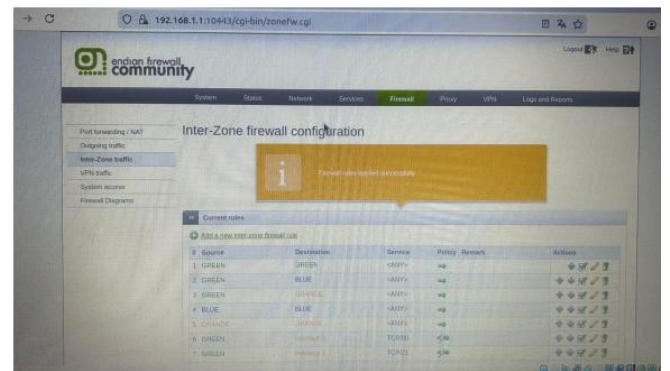
Aplicando Cambios



Fuente: Autoría propia.

Figura 42.

Panel administrativo de Endian Firewall



Fuente: Autoría propia.

Se ingresó al panel administrativo de Endian Firewall Community y en la sección “Firewall → Inter-Zone Traffic” se crearon reglas de acceso entre las zonas RED-GREEN y ORANGE. Posteriormente se configuraron políticas ALLOW para habilitar los servicios HTTP (puerto 80) y FTP (puerto 21), permitiendo la comunicación entre las diferentes zonas de red definidas en el firewall. Finalmente, se aplicaron los cambios mediante la opción “Apply”, verificando el mensaje “Firewall rules applied successfully”, confirmando que las reglas fueron implementadas correctamente.

C. Temática 5: Implementar un Proxy HTTP (No transparente) con políticas de autenticación.

Figura 43.

Creación de Máquina Virtual para Endian

Se hacen algunos cambios en la configuración de red de ambas máquinas virtuales para el funcionamiento con los ejercicios solicitados. Entre estos cambios se ajusta el dispositivo 'Desktop' con 'Red interna' y se asigna la red 'lan-verde' y el dispositivo 'Server' conectado a 'Red interna' y la red 'dmz-naranja'.

Figura 47.

Configuración y asignación de IP en Server.

```
jorge-rincon-server [Condomio] - Oracle VM VirtualBox
jorge-rincon@server:~$ sudo bash -c 'cat < /etc/netplan/00-installer-config.yaml << EOF'
network:
  version: 2
  ethernet:
  enp0s3:
    addresses:
      - 192.168.200.10/24
    routes:
      - to: default
        via: 192.168.200.1
  nameservers:
    addresses: [8.8.8.8, 8.8.4.4]
EOF'
jorge-rincon@server:~$ sudo chmod 600 /etc/netplan/00-installer-config.yaml
jorge-rincon@server:~$ sudo netplan apply
jorge-rincon@server:~$ ip a show enp0s3
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 disc 0 state UP group default qlen 1000
    link/ether 08:00:27:31:6b:81 brd ff:ff:ff:ff:ff:ff
    inet 192.168.200.10/24 brd 192.168.200.255 scope global enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe31:6b81/64 scope link
        valid_lft forever preferred_lft forever
jorge-rincon@server:~$ ping -c 4 192.168.200.1
PING 192.168.200.1 (192.168.200.1) 56(84) bytes of data.
 64 bytes from 192.168.200.1: icmp_seq=1 ttl=64 time=2.75 ms
 64 bytes from 192.168.200.1: icmp_seq=2 ttl=64 time=1.23 ms
 64 bytes from 192.168.200.1: icmp_seq=3 ttl=64 time=1.25 ms
 64 bytes from 192.168.200.1: icmp_seq=4 ttl=64 time=1.01 ms

--- 192.168.200.1 ping statistics ---
 4 packets transmitted, 4 received, 0% packet loss, time 3006ms
 rtt min/avg/max/mdev = 1.010/1.558/2.746/0.692 ms
jorge-rincon@server:~$ _
```

Fuente: Autoría propia.

Con el propósito de integrar un nodo cliente al entorno de red e interconectarlo con la pasarela de seguridad, procedí a realizar la configuración estática del direccionamiento en el servidor virtualizado mediante la herramienta Netplan de Ubuntu. Utilizando privilegios elevados, sobrescribí el archivo /etc/netplan/00-installer-config.yaml para definir los parámetros de la interfaz de red enp0s3, asignándole la dirección IPv4 192.168.200.10/24, la puerta de enlace predeterminada (via: 192.168.200.1) y los servidores de nombres de dominio (DNS de Google: 8.8.8.8 y 8.8.4.4). Posteriormente, restringí los permisos del archivo a través del comando chmod 600 y apliqué los cambios con netplan apply. Finalmente, tras corroborar el direccionamiento activo mediante ip a show enp0s3, ejecuté una prueba de conectividad hacia la puerta de enlace (ping -c 4 192.168.200.1), obteniendo una tasa del 0% de pérdida de paquetes, lo que validó la estabilidad del enlace y la correcta integración del segmento LAN

Figura 48.

Instalación de Squid.

```
jorge-rincon-server [Condomio] - Oracle VM VirtualBox
No containers need to be restarted.
No user sessions are running outdated binaries.
No VM guests are running outdated hypervisor (qemu) binaries on this host.
jorge-rincon@server:~$ sudo bash -c 'cat < /etc/squid/Lists/BlackList.txt << EOF'
jorge-rincon@server:~$ sudo htpasswd -c /etc/squid/Lists/passwd jrincon
New password:
Re-type new password:
Adding password for user jrincon
jorge-rincon@server:~$ cat /etc/squid/Lists/passwd
jrincon:$apr1$9xpRiRIL$jsvzverdB6MndRux13az21
jorge-rincon@server:~$ sudo chown proxy:proxy /etc/squid/Lists/passwd
jorge-rincon@server:~$ sudo chmod 640 /etc/squid/Lists/passwd
jorge-rincon@server:~$ ls -la /etc/squid/Lists/
total 16
drwxr-xr-x 2 root root 4096 May 9 20:41 .
drwxr-xr-x 4 root root 4096 May 9 20:38 ..
-rw-r--r-- 1 root root 87 May 9 20:40 BlackList.txt
-rw-r--r-- 1 proxy proxy 46 May 9 20:41 passwd
jorge-rincon@server:~$ _
```

Fuente: Autoría propia.

Figura 49.

Creación de las listas, el usuario, su contraseña y permisos

```
jorge-rincon-server [Condomio] - Oracle VM VirtualBox
jorge-rincon@server:~$ sudo mkdir -p /etc/squid/Lists
jorge-rincon@server:~$ sudo bash -c 'cat < /etc/squid/Lists/BlackList.txt << EOF'
jorge-rincon@server:~$ cat /etc/squid/Lists/BlackList.txt
hotmail.com
youtube.com
youtube.com
elnuevodia.com.co
elnuevodia.com.co
EOF'
jorge-rincon@server:~$ cat /etc/squid/Lists/BlackList.txt
hotmail.com
youtube.com
youtube.com
elnuevodia.com.co
elnuevodia.com.co
jorge-rincon@server:~$ sudo htpasswd -c /etc/squid/Lists/passwd jrincon
New password:
Re-type new password:
Adding password for user jrincon
jorge-rincon@server:~$ cat /etc/squid/Lists/passwd
jrincon:$apr1$9xpRiRIL$jsvzverdB6MndRux13az21
jorge-rincon@server:~$ sudo chown proxy:proxy /etc/squid/Lists/passwd
jorge-rincon@server:~$ sudo chmod 640 /etc/squid/Lists/passwd
jorge-rincon@server:~$ ls -la /etc/squid/Lists/
total 16
drwxr-xr-x 2 root root 4096 May 9 20:41 .
drwxr-xr-x 4 root root 4096 May 9 20:38 ..
-rw-r--r-- 1 root root 87 May 9 20:40 BlackList.txt
-rw-r--r-- 1 proxy proxy 46 May 9 20:41 passwd
jorge-rincon@server:~$ _
```

Fuente: Autoría propia.

Figura 49.

Configuración de Squid.

```

jorge-rincon-server [Corriendo] - Oracle VirtualBox
jorge-rincon-server:~$ sudo cp /etc/squid/squid.conf /etc/squid/squid.conf.bak
jorge-rincon-server:~$ ls -la /etc/squid/squid.conf
-rw-r--r-- 1 root root 343185 Apr  2 18:27 /etc/squid/squid.conf
jorge-rincon-server:~$ sudo bash -c 'cat' /etc/squid/squid.conf << "EOF"
http_port 3128

auth_param basic program /usr/lib/squid/basic_ncsa_auth /etc/squid/listas/passwd
auth_param basic realm "Proxy UNAD - Autenticacion requerida"
auth_param basic credentialsttl 2 hours
auth_param basic casesensitive off

acl usuarios_autenticados proxy_auth REQUIRED

acl red_local src 192.168.100.0/24
acl SSL_ports port 443
acl Safe_ports port 80
acl Safe_ports port 443
acl Safe_ports port 21
acl Safe_ports port 1025-65535
acl CONNECT method CONNECT

acl sitios_bloqueados dstdomain "/etc/squid/listas/blacklist.txt"

http_access deny !Safe_ports
http_access deny CONNECT !SSL_ports
http_access deny sitios_bloqueados
http_access allow red_local usuarios_autenticados
http_access deny all

cache_mem 256MB
cache_dir ufs /var/spool/squid 1000 16 256
access_log /var/log/squid/access.log squid
cache_log /var/log/squid/cache.log
visible_hostname proxy-unad-ubuntu
EOF
[sudo] password for jorge-rincon:
jorge-rincon-server:~$ _

```

Fuente: Autoría propia.

Figura 50.

Verificación del funcionamiento.

```

jorge-rincon-server [Corriendo] - Oracle VirtualBox
jorge-rincon-server:~$ grep "cache_mem" /etc/squid/squid.conf
cache_mem 256 MB
jorge-rincon-server:~$ sudo squid -k parse
2026/05/09 21:25:06 Startup: Initializing Authentication Schemes ...
2026/05/09 21:25:06 Startup: Initialized Authentication Scheme 'basic'
2026/05/09 21:25:06 Startup: Initialized Authentication Scheme 'digest'
2026/05/09 21:25:06 Startup: Initialized Authentication Scheme 'negotiate'
2026/05/09 21:25:06 Startup: Initialized Authentication Scheme 'ntlm'
2026/05/09 21:25:06 Startup: Initialized Authentication.
2026/05/09 21:25:06 Processing Configuration File: /etc/squid/squid.conf (depth 0)
2026/05/09 21:25:06 Processing: http_port 3128
2026/05/09 21:25:06 Processing: auth_param basic program /usr/lib/squid/basic_ncsa_auth /etc/squid/listas/passwd
2026/05/09 21:25:06 Processing: auth_param basic realm "Proxy UNAD - Autenticacion requerida"
2026/05/09 21:25:06 Processing: auth_param basic credentialsttl 2 hours
2026/05/09 21:25:06 Processing: auth_param basic casesensitive off
2026/05/09 21:25:06 Processing: acl usuarios_autenticados proxy_auth REQUIRED
2026/05/09 21:25:06 Processing: acl red_local src 192.168.100.0/24
2026/05/09 21:25:06 Processing: acl SSL_ports port 443
2026/05/09 21:25:06 Processing: acl Safe_ports port 80
2026/05/09 21:25:06 Processing: acl Safe_ports port 443
2026/05/09 21:25:06 Processing: acl Safe_ports port 21
2026/05/09 21:25:06 Processing: acl Safe_ports port 1025-65535
2026/05/09 21:25:06 Processing: acl CONNECT method CONNECT
2026/05/09 21:25:06 Processing: acl sitios_bloqueados dstdomain "/etc/squid/listas/blacklist.txt"
2026/05/09 21:25:06 Processing: http_access deny !Safe_ports
2026/05/09 21:25:06 Processing: http_access deny CONNECT !SSL_ports
2026/05/09 21:25:06 Processing: http_access deny sitios_bloqueados
2026/05/09 21:25:06 Processing: http_access allow red_local usuarios_autenticados
2026/05/09 21:25:06 Processing: http_access deny all
2026/05/09 21:25:06 Processing: cache_mem 256 MB
2026/05/09 21:25:06 Processing: cache_dir ufs /var/spool/squid 1000 16 256
2026/05/09 21:25:06 Processing: access_log /var/log/squid/access.log squid
2026/05/09 21:25:06 Processing: cache_log /var/log/squid/cache.log
2026/05/09 21:25:06 Processing: visible_hostname proxy-unad-ubuntu
2026/05/09 21:25:06 Initializing https:// proxy context
jorge-rincon-server:~$ _

```

Fuente: Autoría propia.

Para optimizar el tráfico web y aplicar políticas de filtrado de contenido en la red interna, se procedió con la implementación y auditoría de un servidor proxy empleando Squid. Inicialmente, se verifico la asignación de memoria RAM destinada al almacenamiento en caché mediante el comando grep "cache_mem", confirmando un tamaño asignado de 256 MB. Posteriormente, con el fin de garantizar

la integridad operativa del servicio antes de su puesta en marcha, se ejecutó la validación sintáctica del archivo de configuración global a través de la instrucción sudo squid -k parse. La salida detallada en consola demostró que el analizador procesó de forma exitosa los módulos de autenticación básica (ncsa_auth), el esquema de control de acceso (realm 'Proxy UNAD'), las listas de puertos seguros, las restricciones para dominios bloqueados (blacklist.txt) y las jerarquías de reglas de filtrado (http_access), culminando la fase de inicialización sin registrar errores críticos de configuración estructural.

Figura 51.

Comprobación del comportamiento de Squid en Server.

```

jorge-rincon-server [Corriendo] - Oracle VirtualBox
jorge-rincon-server:~$ systemctl stop squid && systemctl status squid | grep -E "Active|Date"
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ====
Authentication is required to stop 'squid.service'.
Authenticating as: jorge-rincon
Password:
==== AUTHENTICATION COMPLETE ====
Active: inactive (dead) since Sat 2026-05-09 21:34:42 UTC; 16s ago
jorge-rincon-server:~$ systemctl start squid && systemctl status squid | grep -E "Active|Date"
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ====
Authentication is required to start 'squid.service'.
Authenticating as: jorge-rincon
Password:
==== AUTHENTICATION COMPLETE ====
Active: active (running) since Sat 2026-05-09 21:35:21 UTC; 30ms ago
jorge-rincon-server:~$ systemctl restart squid && systemctl status squid | grep -E "Active|Date"
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ====
Authentication is required to restart 'squid.service'.
Authenticating as: jorge-rincon
Password:
==== AUTHENTICATION COMPLETE ====
Active: active (running) since Sat 2026-05-09 21:36:17 UTC; 36ms ago
jorge-rincon-server:~$ sudo systemctl enable squid
Synchronizing state of squid.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable squid
jorge-rincon-server:~$ sudo ss -tlnp | grep 3128
LISTEN 0      256          *:*3128          *:*          users:(("squid",p
id=49524,fd=17))
jorge-rincon-server:~$ _

```

Fuente: Autoría propia.

Figura 52.

Conexión LAN establecida entre las máquinas.

```

jorge-rincon-server [Corriendo] - Oracle VirtualBox
Activities Terminal may 10 16:15
jrincon@jorge-rincon-VirtualBox:~$ ping -c 4 192.168.100.1
PING 192.168.100.1 (192.168.100.1) 56(84) bytes of data.
64 bytes from 192.168.100.1: icmp_seq=1 ttl=64 time=1.67 ms
64 bytes from 192.168.100.1: icmp_seq=2 ttl=64 time=0.793 ms
64 bytes from 192.168.100.1: icmp_seq=3 ttl=64 time=0.994 ms
64 bytes from 192.168.100.1: icmp_seq=4 ttl=64 time=1.04 ms

--- 192.168.100.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3006ms
rtt min/avg/max/mdev = 0.793/1.124/1.674/0.330 ms
jrincon@jorge-rincon-VirtualBox:~$ ping -c 4 192.168.200.1
PING 192.168.200.1 (192.168.200.1) 56(84) bytes of data.
64 bytes from 192.168.200.1: icmp_seq=1 ttl=64 time=3.61 ms
64 bytes from 192.168.200.1: icmp_seq=2 ttl=64 time=1.14 ms
64 bytes from 192.168.200.1: icmp_seq=3 ttl=64 time=1.23 ms
64 bytes from 192.168.200.1: icmp_seq=4 ttl=64 time=0.949 ms

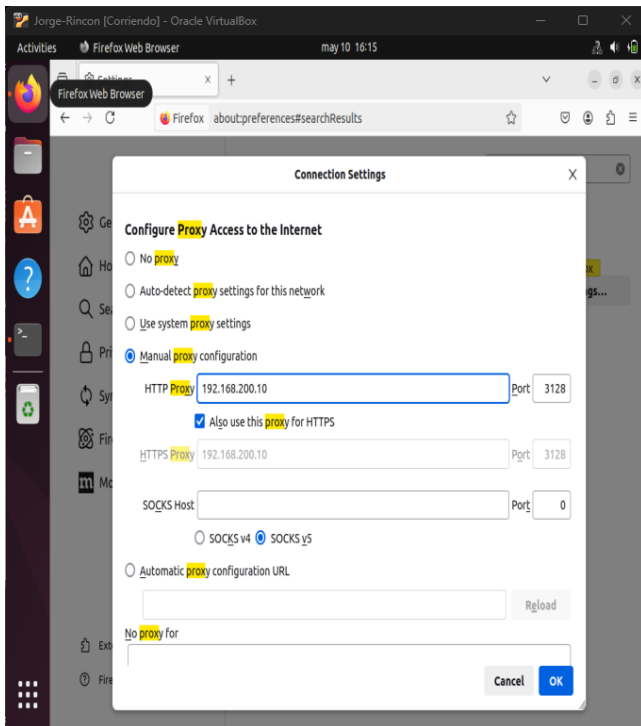
--- 192.168.200.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 0.949/1.729/3.607/1.088 ms
jrincon@jorge-rincon-VirtualBox:~$ _

```

Fuente: Autoría propia.

Figura 53.

Configuración del proxy en Firefox.

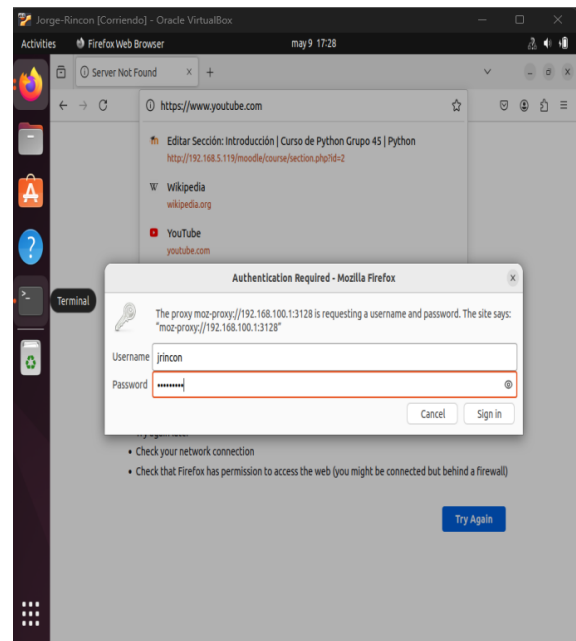


Fuente: Autoría propia.

Para validar la efectividad de las Listas de Control de Acceso (ACL) y las políticas de restricción del servidor proxy, se procedió con la configuración del cliente en la capa de aplicación utilizando el navegador web Mozilla Firefox. Como se ilustra en la ventana de ajustes de conexión (*Connection Settings*), se seleccionó la opción de configuración manual de proxy (*Manual proxy configuration*) se ingreso la dirección IPv4 del servidor Squid (192.168.200.10) junto con el puerto lógico de escucha 3128. Asimismo, se habilito la casilla de verificación para aplicar de manera homóloga estos parámetros al tráfico seguro HTTPS (Also use this proxy for HTTPS), garantizando que todas las solicitudes de navegación perimetral del host final fuesen canalizadas y auditadas por el nodo intermedio.

Figura 54.

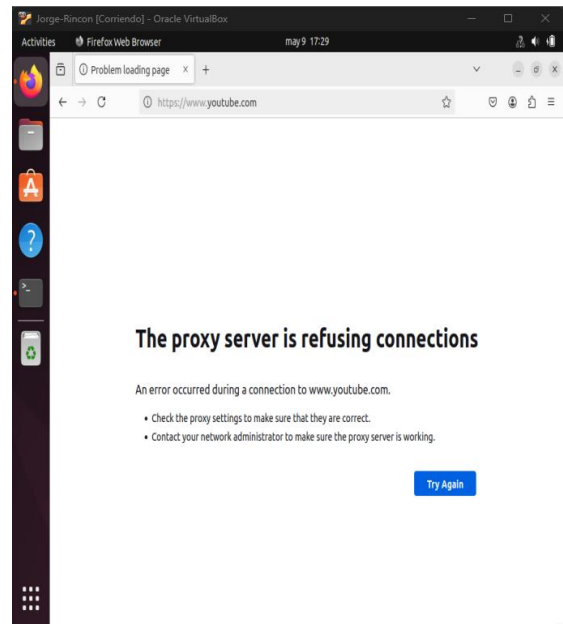
Solicitud de credenciales al ingresar en algún dominio.



Fuente: Autoría propia.

Figura 55.

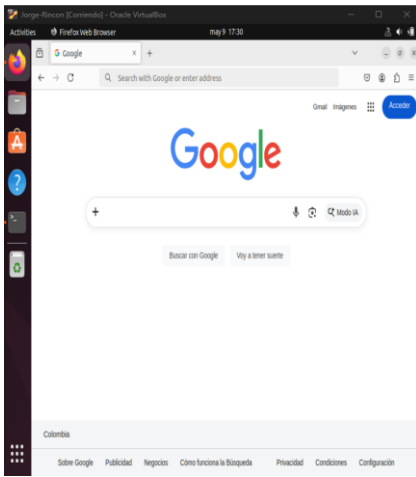
Bloqueo de ingreso a youtube.com



Fuente: Autoría propia.

Figura 56.

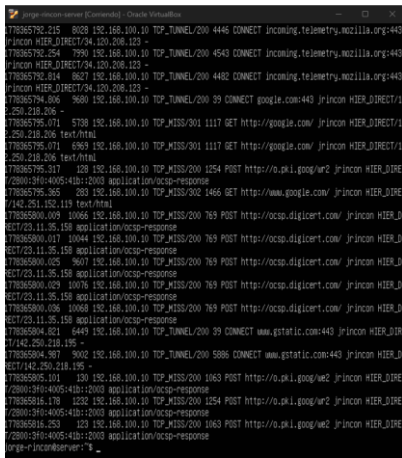
Bloqueo de ingreso a youtube.com



Fuente: Autoría propia.

Figura 57.

Registro de accesos desde el Server.



Fuente: Autoría propia.

I. CONCLUSIONES

La implementación de reglas inter-zona en Endian permitió controlar de forma granular el tráfico entre la LAN y la DMZ, habilitando solo los servicios necesarios (HTTP y FTP) y denegando ICMP.

La práctica evidenció la importancia de mantener un direccionamiento IP estático y persistente en servidores DMZ, para evitar pérdidas de conectividad tras reinicios.

El uso de firewalls basados en zonas como Endian simplifica la administración de políticas de seguridad en entornos GNU/Linux, permitiendo segmentar los recursos críticos de manera eficiente.

Se comprobó que el bloqueo de ping no afecta la comunicación a nivel de aplicación (capa 7), pero agrega una barrera frente a escaneos simples de red que utilizan ICMP

Se logró implementar correctamente la configuración inicial de Endian Firewall

Community en un entorno virtualizado.

- Se configuraron reglas de acceso HTTP y FTP permitiendo la comunicación entre las zonas GREEN y ORANGE.

- La práctica permitió comprender el funcionamiento básico de las políticas de seguridad y control de tráfico en GNU/Linux.

II. REFERENCIAS

LPI LPIC-1 Exam 101. (2022). Tema 101: Determinar y configurar los ajustes de hardware. <https://learning.lpi.org/es/learningmaterials/101-500/101/101.1/>

Hernandez, P. F., & Sánchez, J. (2022). *Monitoreo y administración de sistemas Linux*. [Objeto_virtual_de_información_OVI]. Repositorio Institucional UNAD. <https://repository.unad.edu.co/handle/10596/53211>

Endian. (2016). Endian UTM 3.2 Manual Reference. Recuperado de:

<http://docs.endian.com/3.2/utm/index.html>

Oracle. (2023). VirtualBox User Manual. Recuperado de:

<https://www.virtualbox.org/manual/>

Debian Project. (2024). Debian GNU/Linux Documentation. Recuperado de:

<https://www.debian.org/doc/>

-Linux Professional Institute. (2024). Linux Essentials Learning Materials.

Recuperado de: <https://learning.lpi.org/es/learning-materials/101-500/>