

Implementación de firewall con GNS3 y Oracle VM VirtualBox y Vyos.

Ruiz Tapias Esteban Camilo, Jara Sastoque Jeison, Mendigaño Madero Brayan Nicolás
Universidad Distrital Francisco José de Caldas - Facultad de Ingeniería - Redes de Comunicación II
Bogotá – Colombia
ecruizt@correo.udistrital.edu.co
jejaras@correo.udistrital.edu.co
bnmendiganom@correo.udistrital.edu.co

Resumen— Este proyecto busca realizar la implementación de diferentes sistemas operativos, tanto para uso como dispositivos de borde, como para uso de usuarios finales, se pretende implementar un firewall y la realización de una topología.

Palabras clave: VyOS, GNS3, VirtualBox, Firewall, dispositivo de borde, sistema operativo, webproxy

Abstract - This project seeks to implement different operating systems, both for use as edge devices, and for use by end users, it is intended to implement a firewall and the realization of a topology.

Keywords : VyOS, GNS3, VirtualBox, Firewall, edge device, operating system, webproxy

INTRODUCCIÓN

VyOS es un sistema operativo de red basado en Linux que proporciona enrutamiento de red basado en software, firewall y funcionalidad VPN. En esta oportunidad lo utilizaremos como enrutador y como firewall para una red, además se establecerá un webproxy para limitar los contenidos que pueden consultar los usuarios finales.

I. MARCO DE REFERENCIA

Antes de pasar a la descripción del proyecto y otros apartados veremos algunos conceptos que nos ayudarán a aclarar el funcionamiento de una red.

Así como en la edad media las ciudades amuralladas se defendían de ataques custodiando su perímetro, en la actualidad la defensa contra ataques informáticos se hace custodiando el perímetro de nuestras instalaciones y para ello lo más seguro es que haya solo una puerta de acceso a nuestras redes, de esta forma la red con sus servidores, bases de datos, estaciones de trabajo y demás componentes tendrían un router frontera que conecta la red con el exterior, este será la primera línea de defensa

donde se podrá realizar un filtrado de paquetes utilizando el ACL.

ACL (Access Control List)

ACL o Listas de Control de Acceso, son un conjunto de sentencias que permiten o deniegan un tráfico determinado, tiene diversos usos y es una manera sencilla de filtrar el tráfico que pasa por un dispositivo de red de capa 3, o para el acceso al mismo vía SSH.

SSH (Secure Shell)

Es un protocolo de red que permite el acceso remoto a través de una conexión segura, es decir que por medio de esta conexión se puede administrar el contenido de un servidor, a diferencia de otros protocolos como HTTP, FTP, SSH establece conexiones seguras entre los dos sistemas.

Normalmente la tarea de filtrado de paquetes la realiza el Firewall.

Firewall.

También conocido como cortafuego, es un sistema que protege una computadora o una red de posibles intrusiones que provienen del exterior, generalmente de internet. Este sistema permite filtrar los paquetes de datos que circulan entre la red interna y la externa. Puede ser un programa (software) o un equipo (hardware) que actúa como mediador entre una red local y una o varias redes externas.

La forma más conservadora de filtrar la información que circula hacia la red es bloquear todo el tráfico por defecto, para ir añadiendo reglas para permitir únicamente aquello que deseamos. Los sistemas actúan como firewall se encuentran fuera de la red por lo tanto están expuestas a ataques, estas máquinas se denominan Nodos Bastión por lo que se debe tener cuidado en su configuración teniéndolas lo más seguro que sea posible, con sistemas operativos actualizados, eliminando aplicaciones, protocolos y puertos que no se usen para evitar posibles ataques por medios de estos.

Hay otros servicios en las redes de organizaciones que están expuestos por que deben estar accesibles desde el

exterior como son el servicio web, el correo electrónico, el DNS, entre otros, así que estos servicios se deben proteger con un Firewall y a su vez ser accesibles desde el exterior, por lo que conviene no tenerlos junto con nuestra red interna, por eso se suelen situar en un terreno dentro de la red interna y externa que se suele denominar DMZ.

DMZ (Demilitarized Zone)

En español, zona desmilitarizada, como dijimos se sitúa entre la red interna y la red externa, su función es permitir las conexiones tanto de la red interna como de la externa, facilitando el acceso hacia los servicios mencionados anteriormente pero sin exponer los equipos de la red interna.

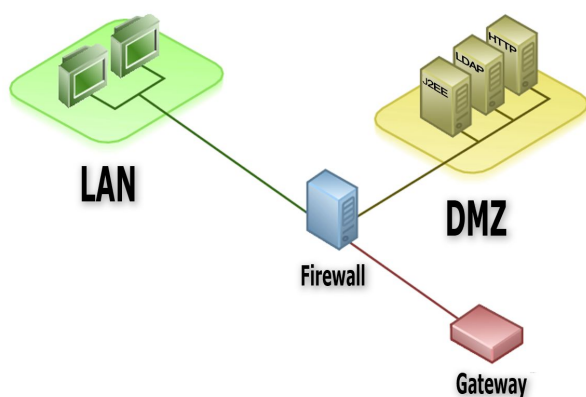


Figura 1. Esquema básico de una red con DMZ, Tomado de [3]

Podemos tratar también otros conceptos que tal vez no sean claros. Cuando se creó internet no se pensó en una red tan extensa como la de hoy en día, por ello se reservó solo 32 bits para las direcciones, que en sí son miles de millones de posibilidades, pero con el aumento exponencial de máquinas conectadas, se agotaban las direcciones IP, para dar una solución, que en verdad podríamos llamarlo “parche”, surgió NAT.

NAT (Network Address translation)

Es español Traducción de direcciones de red, básicamente lo que se busca con esto es hacer que redes de ordenadores hagan uso de un rango de direcciones especiales (IPs privadas) y se conecten a internet usando una sola dirección IP (IP pública), de esta forma por ejemplo una empresa con una gran cantidad de equipos solo utilizara una IP en vez de un por cada equipo.

Network Address Translation - NAT

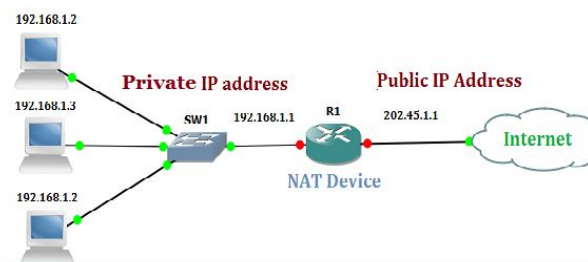


Figura 2. Esquema de funcionamiento de NAT, Tomado de [4]

En un mundo hiperconectado como el de hoy, tener la capacidad de conectar en red dispositivos de forma rápida y fácil es muy valioso, y para ello DHCP es un método esencial para garantizar que los dispositivos puedan conectarse a la red y estén configurados de forma correcta.

DHCP (Dynamic Host Configuration Protocol)

En español protocolo de configuración dinámica de host, es un protocolo que se utiliza en redes IP donde un servidor DHCP asigna automáticamente a cada host una dirección IP, máscara de subred, dirección de puerta de enlace predeterminada, dirección del servicio de nombres de dominio DNS y otros parámetros básicos para que estos host puedan comunicarse de forma correcta con los puntos finales.

Webproxy.

“Un proxy es sencillamente un servidor que gestiona y dirige el tráfico hacia y desde sitios web”[4], cuando los navegadores requieren el servicio de una página se comunican a través del proxy el cual analiza la petición y revisa la fiabilidad de las páginas, e incluso puede detectar software maliciosos.

Para nuestro sistema VyOS el proxy se basa en Squid3 el cual es tiene usos como:

- Aceleración del servidor mediante almacenamiento caché de solicitudes repetidas
- Almacenamiento caché web
- DNS
- Búsquedas de red para computadores que compartan red
- Filtrado de tráfico

[5]

Implementando squidguard para el filtrado de tráfico.

SquidGuard.

La herramienta nos proporciona las siguientes utilidades:

1. Limitar acceso de usuarios a servidores web.
2. Bloquear el acceso a las páginas de la lista negra

3. Bloquear acceso a URL por expresiones o palabras.
4. Redirigir la URL de las paginas bloqueadas.
5. Permitir o negar el uso de direcciones IP en direcciones URL.
6. Redirigir el usuario a un formulario de registro.
7. Redirigir banners a un GIF vacío.
8. Reglas de acceso de fecha y hora.
9. Reglas a usuarios.

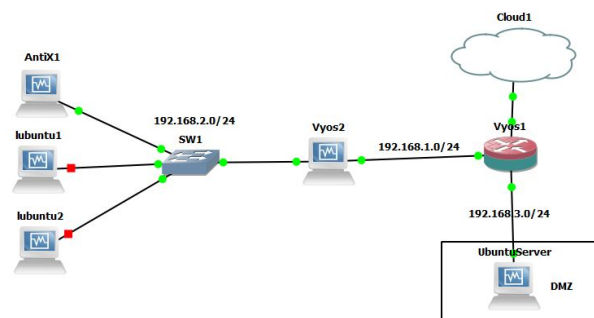


Figura 3. Topología implementada

Para la simulación de nuestra topología, se implementaron los siguientes sistemas operativos en el programa virtualbox:

- Vyos [2]: basado en el sistema operativo de Vyatta core como una herramienta de enrutamiento que proporciona aplicaciones en redes tales como VPNs, firewalls y enrutamiento.
- AntiX: es una distribución de Linux construida directamente sobre Debian. Es comparativamente liviano y adecuado para computadoras más antiguas, al mismo tiempo que proporciona kernel y aplicaciones de vanguardia, así como actualizaciones y adiciones a través del sistema de paquetes apt-get y repositorios compatibles con Debian.
- Lubuntu: es una distribución Linux ligera, basada en Ubuntu, que emplea el entorno de escritorio LXQt en lugar del entorno GNOME de Ubuntu.

III. IMPLEMENTACIÓN

En esta ocasión pretendemos hacer una implementación de:

- Webproxy
- Servidor WEB
- Firewall
- DMZ
- Acceso SSH

Para ello utilizamos la siguiente topología básica:

Teniendo en cuenta esta topología procedemos a realizar la configuración del VyOS1 que será el encargado además del firewall y de la conexión a internet.

Para poder probar el funcionamiento del webproxy se requiere la conexión a internet en este caso, se habilita una red compartida con un adaptador de bucle invertido el cual llamaremos loopback, se habilita en la topología de GNS3, después se coloca una primera VM con VyOS1, que maneja la conexión a internet y que llevará la regla nat que comunicara la red LAN con la WAN.

Para la conexión WAN se configura DHCP y se asigna una red a las demás interfaces del dispositivo, luego hacemos un enrutamiento estático para que se pueda dar la comunicación.

El VyOS2 será el encargado de manejar el webproxy que se aplica a la red 192.168.2.0 limitando el contenido que se puede navegar en los computadores de dicha red, también se configura un DHCP para generar los dispositivos de esta red.

En el apartado de la DMZ configuramos un servidor web permitiendo las conexiones tanto desde la red interna como de la externa, mientras que las conexiones que parten de la DMZ solo puedan salir a la red interna.

La configuración completa de estos dispositivos se puede encontrar en el github del proyecto TopologiaProxyFireWallVyOS [7]

Mediante la implementación de estos dispositivos se generó la siguiente tabla de direcciones IP:

Equipo	IP	Máscara	Gateway
vyos1	dhcp	0.0.0.255	192.168.137.1
	192.168.1.1	0.0.0.255	-
	192.168.3.1	0.0.0.255	
vyos2	192.168.1.2	0.0.0.255	-
	192.168.2.1	0.0.0.255	-

Antix1	dhcp	0.0.0.255	192.168.2.1
lubuntu1	dhcp	0.0.0.255	192.168.2.1
lubuntu2	dhcp	0.0.0.255	192.168.2.1
ubuntuserver	192.168.3.2	0.0.0.255	192.168.3.1

Tabla 1. Direcciones ip de la topología

IV. FIREWALL

En línea de comandos podemos aceptar, soltar y rechazar paquetes de diferentes protocolos, además de especificar el tipo de tráfico que se requiere gestionar, teniendo tráfico LOCAL, IN y OUT; los cuales especifican el tráfico dentro del sistema VyOS, tráfico entrante y saliente.

Reglas	Grupos				Firewall		
	NET DMZ	NET LAN	NET MANAGEMENT	NET WAN	WAN LOCAL	LAN LOCAL	DMZ LOCAL
1010					X	X	X
1011					X	X	X
1020					X	X	X
1030						X	
1040						X	X
1100			X			X	X
1101			X			X	X
1110			X		X	X	X

Tabla 2. Reglas aplicadas en el firewall.

Para la configuración del firewall se tienen en cuenta los siguientes estados:

- accept
- established
- related
- drop
- invalid

Configuración	Reglas							
	1010	1011	1020	1030	1040	1100	1101	1110
action	accept	drop		accept		drop	accept	accept

established	enable							
related	enable							
invalid		enable						
icmp type name			echo request					
protocol			icmp	udp	tcp udp	tcp	tcp	udp
new			enable	enable	enable	enable	enable	enable
destination port				67	53	22	22	161
recent count						4		
recent time						60		

Tabla 3. Breve explicación de las reglas

aplicadas.

V. PRUEBAS Y SIMULACIÓN

A continuación se muestra como se encuentran configurados cada uno de los dispositivos y los diferentes resultados de las conexiones.

vyos1

Puertos de ethernet:

```
vyos@vyos1:~$ show interfaces
Codes: S - State, L - Link, u - Up, D - Down, A - Admin Down
Interface      IP Address      S/L  Description
-----
eth0           192.168.137.111/24  u/u   red compartida
eth1           192.168.1.1/24    u/u   LAN
eth2           192.168.3.1/24    u/u   DMZ
eth3           -                 u/D
lo             127.0.0.1/8       u/u
::1/128
```

Conexión a internet:

```
vyos@vyos1:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_req=1 ttl=117 time=91.9 ms
64 bytes from 8.8.8.8: icmp_req=2 ttl=117 time=9.97 ms
64 bytes from 8.8.8.8: icmp_req=3 ttl=117 time=80.1 ms
^C
--- 8.8.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 9.971/60.672/91.924/36.174 ms
vyos@vyos1:~$
```

vyos2

Puertos de ethernet:

```
vyos@vyos2:~$ show interfaces
Codes: S - State, L - Link, u - Up, D - Down, A - Admin Down
Interface      IP Address      S/L  Description
-----
eth0            192.168.1.2/24  u/u
eth1            192.168.2.1/24  u/u
eth2            -              u/D
eth3            -              u/D
lo              127.0.0.1/8    u/u
:::1/128
```

Conexión a internet:

```
vyos@vyos2:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_req=1 ttl=116 time=123 ms
64 bytes from 8.8.8.8: icmp_req=2 ttl=116 time=52.7 ms
64 bytes from 8.8.8.8: icmp_req=3 ttl=116 time=125 ms
^C
--- 8.8.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 52.703/100.674/125.815/33.934 ms
vyos@vyos2:~$
```

UbuntuServer

Conexión a internet:

```
ubuntuuser@ubuntuuser:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=116 time=117 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=116 time=260 ms
^C
--- 8.8.8.8 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 117.350/188.754/260.158/71.404 ms
ubuntuuser@ubuntuuser:~$
```

Apache2:

```
ubuntuuser@ubuntuuser:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2020-08-20 19:41:09 UTC; 17min ago
     Docs: https://httpd.apache.org/docs/2.4/
   Process: 725 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
    Main PID: 774 (apache2)
       Tasks: 55 (limit: 573)
      Memory: 7.6M
      CGroup: /system.slice/apache2.service
              └─774 /usr/sbin/apache2 -k start
                └─776 /usr/sbin/apache2 -k start
                  └─777 /usr/sbin/apache2 -k start

Aug 20 19:41:09 ubuntuuser systemd[1]: Starting The Apache HTTP Server...
Aug 20 19:41:09 ubuntuuser apachectl[743]: AH00558: apache2: Could not reliably determine the server's fully qualified domain name, please add the 'ServerName' directive to the configuration to avoid this warning.
Aug 20 19:41:09 ubuntuuser systemd[1]: Started The Apache HTTP Server.
ubuntuuser@ubuntuuser:~$
```

Control SSH:

```
ubuntuuser@ubuntuuser:~$ ssh 192.168.3.1
ssh: connect to host 192.168.3.1 port 22: Connection timed out
ubuntuuser@ubuntuuser:~$
```

Antix

IP:

```
$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST,DYNAMIC> mtu 1500
    inet 192.168.2.3 netmask 255.255.255.0 broadcast 192.168.2.255
    inet6 fe80::a00:27ff:fe5a:8095 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:5a:80:95 txqueuelen 1000 (Ethernet)
    RX packets 4668 bytes 3579354 (3.4 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 2389 bytes 259097 (253.0 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1 (Local Loopback)
    RX packets 428 bytes 40202 (39.2 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 428 bytes 40202 (39.2 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

antix1@antix1:~$
```

Conexión a internet:

```
antix1@antix1:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=115 time=115 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=115 time=40.5 ms
^C
--- 8.8.8.8 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 3ms
rtt min/avg/max/mdev = 40.535/77.851/115.168/37.317 ms
antix1@antix1:~$
```

Conexión con el servidor:

The screenshot shows a web browser window with the address bar displaying '192.168.3.2'. The page content includes a heading 'Conexión exitosa con el servidor!' and a list of pages. The list contains two items: 'Servidor: [apache2](#)' and 'Paginas bloqueadas: [block](#)'. In the background, a terminal window is visible, showing a successful ping to 192.168.3.2 and a successful SSH connection to the same IP.

Control SSH:

```
antix1@antix1:~$ ssh vyos@192.168.1.1
Welcome to VyOS
vyos@192.168.1.1's password:
Linux vyos1 3.13.11-1-amd64-vyos #1 SMP Sat Nov 11 12:10:30 CET 2017 x86_64
Welcome to VyOS.
This system is open-source software. The exact distribution terms for
each module comprising the full system are described in the individual
files in /usr/share/doc/*/copyright.
Last login: Thu Aug 20 19:51:04 2020
vyos@vyos1:~$
```

Lubuntu1

IP:


```

lubuntu1@lubuntu1:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.2.2 netmask 255.255.255.0 broadcast 192.168.2.255
    inet6 fe80::a00:27ff:fe28:6556 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:28:65:56 txqueuelen 1000 (Ethernet)
    RX packets 448 bytes 476632 (476.6 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 344 bytes 36284 (36.2 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 426 bytes 29978 (29.9 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 426 bytes 29978 (29.9 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lubuntu1@lubuntu1:~$

```

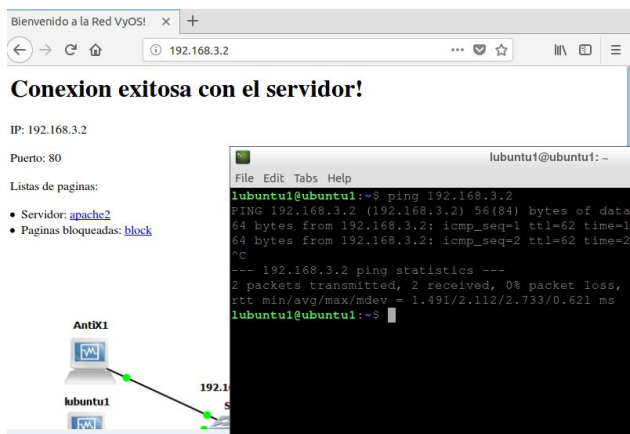
Conexión internet:

```

lubuntu1@lubuntu1:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data:
64 bytes from 8.8.8.8: icmp_seq=1 ttl=115 time=11.8 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=115 time=27.5 ms
^C
--- 8.8.8.8 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 100lms
rtt min/avg/max/mdev = 11.847/19.681/27.516/7.835 ms
lubuntu1@lubuntu1:~$

```

Conexión con el servidor:



Control SSH:

```

lubuntu1@lubuntu1:~$ ssh vyos@192.168.3.1
Welcome to VyOS
vyos@192.168.3.1's password:
Linux vyos1 3.13.11-l-amd64-vyos #1 SMP Sat Nov 11 12:10:30 CET 2017 x86_64
Welcome to VyOS.
This system is open-source software. The exact distribution terms for
each module comprising the full system are described in the individual
files in /usr/share/doc/*/copyright.
Last login: Thu Aug 20 20:05:10 2020 from 192.168.2.3
vyos@vyos1:~$

```

lubuntu2

Se debe tener en cuenta que lubuntu 2 es clonado de lubuntu 1, por lo que cuentan con el mismo usuario.

IP:

```

lubuntu1@lubuntu1:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.2.6 netmask 255.255.255.0 broadcast 192.168.2.255
    inet6 fe80::a00:27ff:fe97:7425 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:97:74:25 txqueuelen 1000 (Ethernet)
    RX packets 150 bytes 92921 (92.9 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 198 bytes 26614 (26.6 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 297 bytes 22907 (22.9 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 297 bytes 22907 (22.9 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lubuntu1@lubuntu1:~$

```

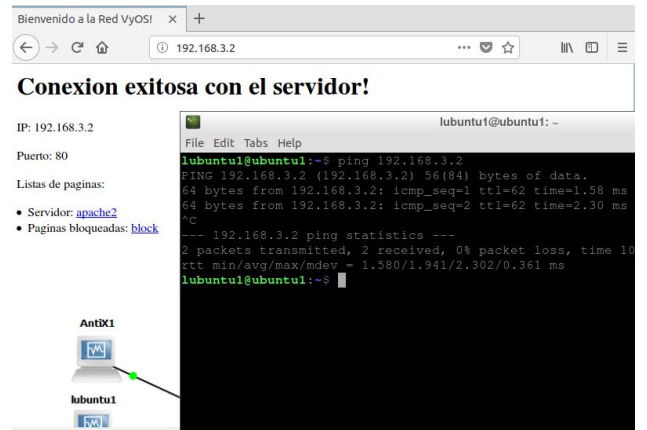
Conexión internet:

```

lubuntu1@lubuntu1:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data:
64 bytes from 8.8.8.8: icmp_seq=1 ttl=115 time=20.2 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=115 time=106 ms
^C
--- 8.8.8.8 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 100lms
rtt min/avg/max/mdev = 20.266/63.413/106.561/43.148 ms
lubuntu1@lubuntu1:~$

```

Conexión con el servidor:



Control SSH:

```

lubuntu1@lubuntu1:~$ ssh vyos@192.168.1.1
ssh: connect to host 192.168.1.1 port 22: Connection timed out
lubuntu1@lubuntu1:~$

```

CONCLUSIONES

- El manejo de VyOS es interesante ya que no se cuenta con una interfaz gráfica que pueda facilitar la implementación de configuración, por lo que lo hace muy similar a la manipulación de la terminal de linux shell.
- VyOS es un software bastante moldeable de acuerdo a las necesidades que se pretendan implementar, sin embargo tiene algunas limitaciones respecto a sus últimas actualizaciones, y la documentación no es tan completa.
- La zona DMZ sirve para utilizar cierta clase de servicios sin poner en riesgo la red interna de una topología, pero esta herramienta solo es eficiente cuando se tiene un firewall que acompañe el acceso a la zona DMZ.
- El Webproxy nos ayuda a limitar el contenido que puede consultar los usuarios de una red sin necesidad de llegar a manipular su dispositivo, es útil si queremos llegar a implementar un control a los usuarios finales de una red. En este caso fue un poco limitado.

- El Secure Schell (SSH) permite el acceso remoto a un servidor por medio de un canal seguro en el que toda la información está cifrada, esto es conveniente si se necesita otorgar permisos a personas para que puedan modificar un servidor.
-

REFERENCIAS

- [1] u/mianosm (2014) Realise VyOS 1.00 [online] disponible en:
https://www.reddit.com/r/networking/comments/1thfaw/release_vyos_100_an_enhanced_fork_based_from_the/
- [2] “DMZ, Toda la información que necesitas para crear tu propia zona desmilitarizada” [Online] Disponible en:
<https://www.androidzte.com/dmz-toda-la-informacion-que-necesitas-para-crear-tu-propia-zona-desmilitarizada/>. Visitado 15 Ago de 2020
- [3] “Direccionamiento de puertos dst-nat destination network address translation”. [Online]. Disponible en:
<https://danilopy.wordpress.com/2017/04/03/direccionamiento-de-puertos-dst-nat-distination-network-address-translation/>
- [4] Ricardi Saenz (26, julio, 2016) ¿Qué es realmente un servidor proxy? [online] disponible en:
<https://icloudseven.com/que-es-realmente-un-web-proxy/>
- [5] VyOS Platform Repositories, Documentación [online] disponible en:
<https://docs.vyos.io/en/latest/>
- [6] Christine Kronberg, About SquirdGuard [online] disponible en:
<http://www.squidguard.org/about.html>
- [7]

<https://enredandoconredes.com/2015/01/08/acls-listas-de-control-de-acceso/>

<https://desafiohosting.com/que-es-ssh/>