

Monitorizar el sistema

Al ejecutar el comando `top` en nuestra terminal nos mostrara una tabla, algunos de los encabezados son:

1. **PID (Process ID)**: Es el identificador único asignado a cada proceso en el sistema. Te permite identificar un proceso específico.
2. **Usuario (USER)**: El nombre del usuario propietario del proceso.
3. **% CPU (CPU Usage)**: El porcentaje de tiempo de CPU que está utilizando cada proceso desde la última actualización de **top**.
4. **% MEM (Memory Usage)**: El porcentaje de memoria RAM utilizado por cada proceso.
5. **Comando (Command)**: El comando o programa que está siendo ejecutado por el proceso.
6. **Tiempo de CPU (CPU Time)**: El tiempo total de CPU que ha utilizado el proceso desde que se inició.
7. **Prioridad (Priority)**: La prioridad del proceso en el sistema.
8. **NI (Nice value)**: El valor "nice" asignado al proceso, que afecta su prioridad de planificación.
9. **%CPU (CPU Utilization)**: El porcentaje de uso de la CPU global del sistema.
10. **%MEM (Memory Utilization)**: El porcentaje de uso de memoria RAM global del sistema.

```
top - 16:45:56 up 16 min, 1 user, load average: 0.03, 0.07, 0.03
Tasks: 185 total, 1 running, 184 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.1 us, 0.2 sy, 0.0 ni, 99.7 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 3919.3 total, 2645.6 free, 625.3 used, 648.4 buff/cache
MiB Swap: 975.0 total, 975.0 free, 0.0 used, 3063.2 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1214	agus	20	0	4435620	264068	122032	S	0.7	6.6	0:18.27	gnome-shell
506	root	20	0	0	0	0	I	0.3	0.0	0:00.30	kworker/2:4-events
1541	agus	20	0	217520	2404	2052	S	0.3	0.1	0:01.12	VBoxClient
2140	root	20	0	10152	3920	3192	R	0.3	0.1	0:00.12	top
1	root	20	0	164080	10420	7804	S	0.0	0.3	0:01.30	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kthreadd
3	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_gp
4	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_par_gp
6	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/0:0H-events_
8	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	mm_percpu_wq
9	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcu_tasks_rude_
10	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcu_tasks_trace
11	root	20	0	0	0	0	S	0.0	0.0	0:00.05	ksoftirqd/0
12	root	20	0	0	0	0	I	0.0	0.0	0:00.46	rcu_sched
13	root	rt	0	0	0	0	S	0.0	0.0	0:00.03	migration/0
14	root	20	0	0	0	0	I	0.0	0.0	0:00.03	kworker/0:1-cgroup_d
15	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/0
16	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/1
17	root	rt	0	0	0	0	S	0.0	0.0	0:00.29	migration/1
18	root	20	0	0	0	0	S	0.0	0.0	0:00.11	ksoftirqd/1

Analizando procesos

En sistemas basados en Linux, incluyendo Debian, **systemd** es un sistema de inicialización y gestión de procesos que ha reemplazado al tradicional SysV init. Fue diseñado para mejorar el proceso de arranque del sistema y gestionar servicios y procesos en tiempo de ejecución. **systemd** ha sido ampliamente adoptado en muchas distribuciones de Linux debido a sus ventajas y características.

1. **systemd** se compone de varios componentes, y algunos de los principales son:
 - 1.1. **systemd init**: Es el proceso inicial (PID 1) que se encarga de iniciar todos los demás procesos y servicios durante el arranque del sistema.
 - 1.2. **systemd units (unidades)**: Son archivos de configuración que describen servicios, dispositivos, puntos de montaje, sockets y otros recursos gestionados por **systemd**. Cada unidad representa un servicio o recurso específico que **systemd** puede controlar.
 - 1.3. **systemd service**: Representa un servicio que se puede administrar con **systemd**. Estas unidades describen cómo iniciar, detener, reiniciar y gestionar procesos en el sistema.
 - 1.4. **systemd target**: Es un grupo de unidades que se activan o desactivan juntas. Por ejemplo, hay objetivos específicos para el modo de usuario único (single-user.target) o el modo multiusuario con GUI (graphical.target).
 - 1.5. **systemd journald**: Es el componente responsable de la recopilación y gestión de registros (logs) del sistema.
 - 1.6. **systemd logind**: Se encarga de la gestión de sesiones de usuario, el control de la energía y la administración de dispositivos.
 - 1.7. **systemd udev**: Se utiliza para la detección y administración dinámica de dispositivos.

2. **gnome-shell**: Es el proceso principal del entorno de escritorio GNOME. Se encarga de administrar la interfaz gráfica del usuario, proporcionando la barra superior, el lanzador de aplicaciones, la administración de ventanas y otros componentes del escritorio.
3. **kworker/2:4-events**: Los procesos que comienzan con **kworker** son parte del kernel de Linux y generalmente están relacionados con trabajadores de kernel que realizan tareas en segundo plano. El número después de **kworker/** (en este caso, **2:4**) indica el número de CPU y la tarea específica que está realizando el trabajador del kernel.
4. **kthreadd**: Es el primer hilo (thread) creado por el kernel de Linux y actúa como el "padre" de todos los demás hilos en el sistema. Su función principal es crear y destruir otros hilos del kernel según sea necesario.
5. **rcu_gp**: RCU (Read-Copy-Update) es un mecanismo utilizado en el kernel de Linux para permitir el acceso concurrente a estructuras de datos compartidas sin bloqueos. **rcu_gp** representa el subsistema de Protección de Grupo RCU, que es responsable de administrar el mecanismo RCU.
6. **mm_percpu_wq**: Este proceso está relacionado con el subsistema de manejo de memoria del kernel de Linux. El "mm" en el nombre se refiere a "Memory Management" (Gestión de Memoria). **percpu** significa "per-CPU" (por CPU), lo que indica que está relacionado con el manejo de la memoria asignada a cada núcleo del procesador. "wq" representa "workqueue" (cola de trabajo), que es una estructura utilizada para programar tareas asíncronas en el kernel.

Net-tools

Este comando proporciona información útil para diagnosticar problemas de red y monitorear el rendimiento del sistema. A continuación, se enumeran algunas de las estadísticas que se pueden obtener con **netstat -s**:

1. **Estadísticas TCP y UDP**: Proporciona información sobre la cantidad de paquetes enviados y recibidos, errores en la transmisión y retransmisiones para los protocolos TCP y UDP.
2. **Estadísticas ICMP**: Muestra estadísticas relacionadas con el Protocolo de Mensajes de Control de Internet (ICMP), que se utiliza para mensajes de error y control en la capa de red.
3. **Estadísticas IP**: Ofrece detalles sobre el tráfico IP, incluyendo paquetes recibidos, enviados, descartados y errores en el envío y recepción.
4. **TCP (Transmission Control Protocol)**: Proporciona estadísticas relacionadas con el tráfico y la actividad de la capa de transporte utilizando el protocolo TCP. Incluye información sobre el número de segmentos enviados y recibidos, conexiones establecidas y finalizadas, retransmisiones, errores de conexión, entre otros.
5. **UDP (User Datagram Protocol)**: Muestra estadísticas relacionadas con el tráfico y la actividad de la capa de transporte utilizando el protocolo UDP. Esto incluye información sobre el número de datagramas enviados y recibidos, errores de envío, errores de recepción, entre otros.
6. **tcpExt**: Representa un conjunto de estadísticas extendidas relacionadas con el protocolo TCP. Incluye estadísticas más detalladas sobre la actividad de TCP, como conexiones y segmentos rechazados, diferentes tipos de errores, tiempos de espera, entre otros.

7. **IPExt:** Representa un conjunto de estadísticas extendidas relacionadas con el protocolo IP (Internet Protocol). Incluye información detallada sobre la actividad de la capa de red, como paquetes enviados, paquetes descartados, errores en el envío, errores en la recepción, entre otros.

```
root@Debian:~# netstat -s
Ip:
    Forwarding: 2
    7377 total packets received
    1 with invalid addresses
    0 forwarded
    0 incoming packets discarded
    7374 incoming packets delivered
    3631 requests sent out
Icmp:
    0 ICMP messages received
    0 input ICMP message failed
    ICMP input histogram:
    17 ICMP messages sent
    0 ICMP messages failed
    ICMP output histogram:
        destination unreachable: 17
IcmpMsg:
    OutType3: 17
Tcp:
    65 active connection openings
    0 passive connection openings
```

Tcpdump

El comando "tcpdump -i eth0" se utiliza para capturar y mostrar el tráfico de red en una interfaz de red específica, en este caso, "eth0". Tcpdump es una herramienta de línea de comandos que está disponible en sistemas operativos basados en Unix/Linux y se utiliza principalmente para analizar y depurar el tráfico de red. Al ejecutar "tcpdump -i eth0", el comando escuchará el tráfico que fluye a través de la interfaz "eth0" y mostrará en tiempo real los paquetes capturados en la terminal. Cada línea de salida representa un paquete capturado y proporciona información detallada sobre los campos de cabecera del paquete, como dirección IP de origen y destino, puertos TCP/UDP, etc.

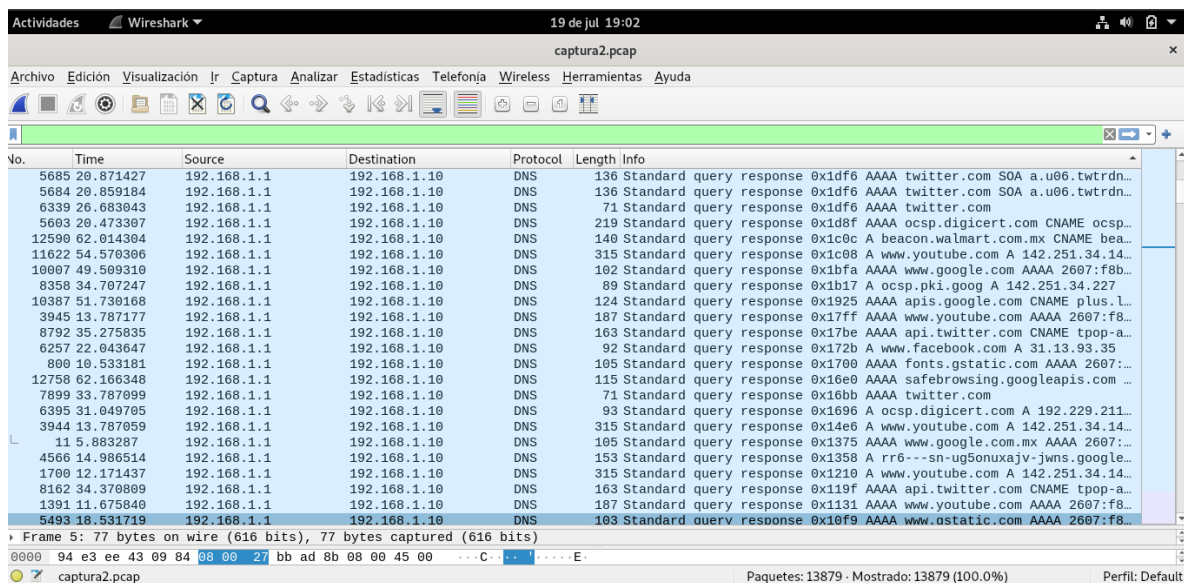
Este comando capturará el tráfico en "eth0" y lo guardará en un archivo llamado "captura.pcap". Luego, puedes utilizar herramientas como Wireshark para analizar el contenido del archivo capturado y examinar los paquetes en detalle.

```
root@Debian:~# tcpdump -i enp0s3 -w captura2.pcap
tcpdump: listening on enp0s3, link-type EN10MB (Ethernet), snapshot length 262144 bytes
^C13879 packets captured
13937 packets received by filter
0 packets dropped by kernel
root@Debian:~# mv captura2.pcap /home/
```

Al abrir el archivo que creamos en wireshark podemos analizar el tráfico de paquetes que fueron enviados y recibidos, algunos datos importantes en los que se muestran son las direcciones IP de origen y destino, el protocolo, el tamaño del paquete y la información de dicho paquete. Al

capturar el tráfico de red, puedes obtener una visión detallada de las comunicaciones que ocurren en tu red, lo que puede ser útil para diversos propósitos:

1. **Diagnóstico de problemas de red:** Capturar el tráfico te permite analizar las comunicaciones entre dispositivos para identificar problemas de rendimiento, fallos, cuellos de botella y otros problemas de red. Esto es especialmente útil cuando los usuarios experimentan lentitud o desconexiones inesperadas.
2. **Monitoreo del tráfico de red:** Esto es útil para comprender el uso de la red, identificar comportamientos anómalos o tráfico no autorizado y supervisar la eficiencia de los servicios.
3. **Seguridad y análisis de amenazas:** La captura del tráfico de red es una herramienta esencial para investigar incidentes de seguridad, intrusiones, malware o actividades maliciosas. Permite analizar patrones y comportamientos sospechosos, así como detectar intentos de acceso no autorizado a recursos de red.
4. **Verificación de cumplimiento:** En entornos empresariales y organizacionales, es importante garantizar que se cumplan las políticas de seguridad y privacidad. La captura del tráfico de red puede ayudar a verificar si los empleados siguen las pautas y si no se están transmitiendo datos confidenciales de forma no autorizada.
5. **Desarrollo y pruebas de aplicaciones:** Los desarrolladores de software y administradores de sistemas pueden utilizar la captura del tráfico para diagnosticar problemas en aplicaciones, depurar problemas de comunicación y evaluar el rendimiento de los servicios en tiempo real.



The screenshot shows the Wireshark interface with a packet capture of DNS traffic. The packet list on the left shows a series of DNS queries and responses between 192.168.1.1 and 192.168.1.10. The selected packet (No. 5493) is a DNS response from 192.168.1.10 to 192.168.1.1. The packet details pane on the right shows the structure of the DNS response, including the header, question, answer, and authority sections. The packet bytes pane at the bottom shows the raw data of the packet.

No.	Time	Source	Destination	Protocol	Length	Info
5685	20.871427	192.168.1.1	192.168.1.10	DNS	136	Standard query response 0x1df6 AAAA twitter.com SOA a.u06.twtrdn...
5684	20.859184	192.168.1.1	192.168.1.10	DNS	136	Standard query response 0x1df6 AAAA twitter.com SOA a.u06.twtrdn...
6339	26.683043	192.168.1.1	192.168.1.10	DNS	71	Standard query response 0x1df6 AAAA twitter.com
5603	20.473307	192.168.1.1	192.168.1.10	DNS	219	Standard query response 0x1d8f AAAA ocsp.digicert.com CNAME ocsp...
12590	62.014304	192.168.1.1	192.168.1.10	DNS	140	Standard query response 0x1c0c A beacon.walmart.com.mx CNAME bea...
11622	54.570306	192.168.1.1	192.168.1.10	DNS	315	Standard query response 0x1c08 A www.youtube.com A 142.251.34.14...
10007	49.509310	192.168.1.1	192.168.1.10	DNS	102	Standard query response 0x1bfa AAAA www.google.com AAAA 2607:f8b...
8358	34.707247	192.168.1.1	192.168.1.10	DNS	89	Standard query response 0x1b17 A ocsp.pki.goog A 142.251.34.227
10387	51.730168	192.168.1.1	192.168.1.10	DNS	124	Standard query response 0x1925 AAAA apis.google.com CNAME plus.l...
3945	13.787177	192.168.1.1	192.168.1.10	DNS	187	Standard query response 0x17ff AAAA www.youtube.com AAAA 2607:f8...
8792	35.275835	192.168.1.1	192.168.1.10	DNS	163	Standard query response 0x17be AAAA api.twitter.com CNAME tpop-a...
6257	22.043647	192.168.1.1	192.168.1.10	DNS	92	Standard query response 0x172b A www.facebook.com A 31.13.93.35
800	10.533181	192.168.1.1	192.168.1.10	DNS	105	Standard query response 0x1700 AAAA fonts.gstatic.com AAAA 2607:f8...
12758	62.166348	192.168.1.1	192.168.1.10	DNS	115	Standard query response 0x16e0 AAAA safebrowsing.googleapis.com ...
7899	33.787099	192.168.1.1	192.168.1.10	DNS	71	Standard query response 0x16bb AAAA twitter.com
6395	31.049705	192.168.1.1	192.168.1.10	DNS	93	Standard query response 0x1696 A ocsp.digicert.com A 192.229.211...
3944	13.787059	192.168.1.1	192.168.1.10	DNS	315	Standard query response 0x14e6 A www.youtube.com A 142.251.34.14...
11	5.883287	192.168.1.1	192.168.1.10	DNS	105	Standard query response 0x1375 AAAA www.google.com.mx AAAA 2607:f8...
4566	14.986514	192.168.1.1	192.168.1.10	DNS	153	Standard query response 0x1358 A rr6---sn-ug5onuxajv-jwns.google...
1700	12.171437	192.168.1.1	192.168.1.10	DNS	315	Standard query response 0x1210 A www.youtube.com A 142.251.34.14...
8162	34.370809	192.168.1.1	192.168.1.10	DNS	163	Standard query response 0x119f AAAA api.twitter.com CNAME tpop-a...
1391	11.675840	192.168.1.1	192.168.1.10	DNS	187	Standard query response 0x1131 AAAA www.youtube.com AAAA 2607:f8...
5493	18.531719	192.168.1.1	192.168.1.10	DNS	103	Standard query response 0x10f9 AAAA www.gstatic.com AAAA 2607:f8...

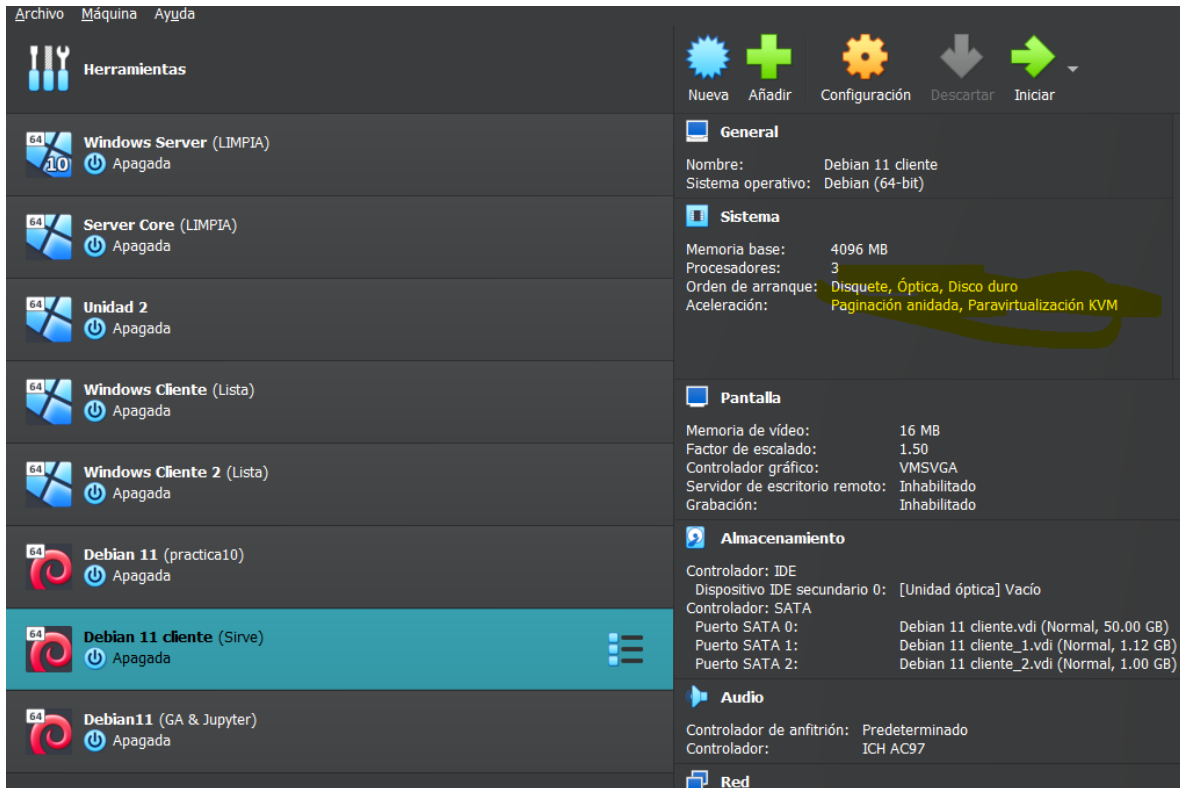
Frame 5: 77 bytes on wire (616 bits), 77 bytes captured (616 bits)

0000 94 e3 ee 43 09 84 08 00 27 bb ad 0b 08 00 45 00 ...C...E

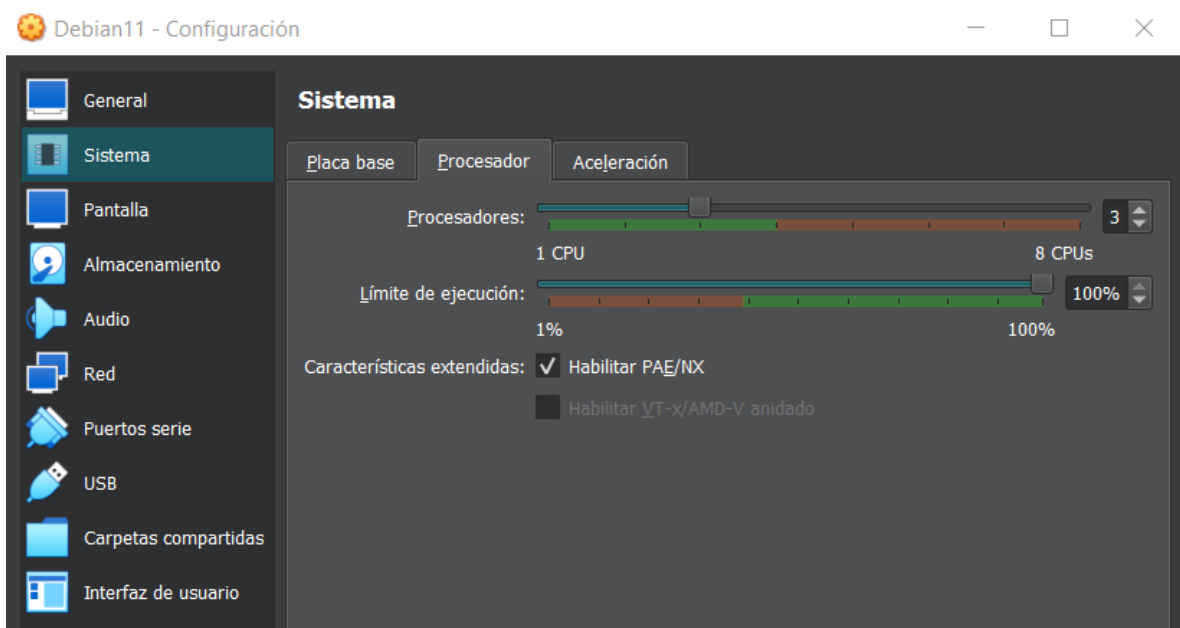
captura2.pcap Paquetes: 13879 - Mostrado: 13879 (100.0%) Perfil: Default

instalación de DHCP

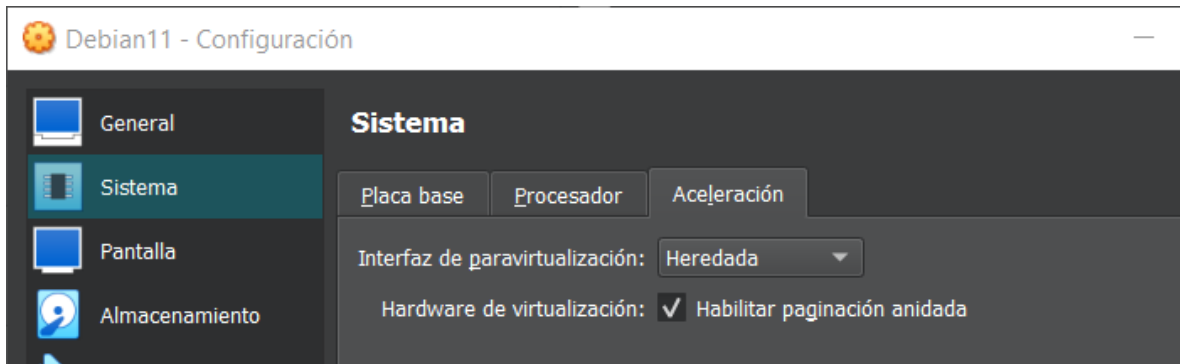
Para poder utilizar correctamente los adaptadores de red, debemos tener nuestro servidor y cliente de manera virtualizada, en caso de tenerlas paravirtualizadas debemos modificar este tipo de virtualización.



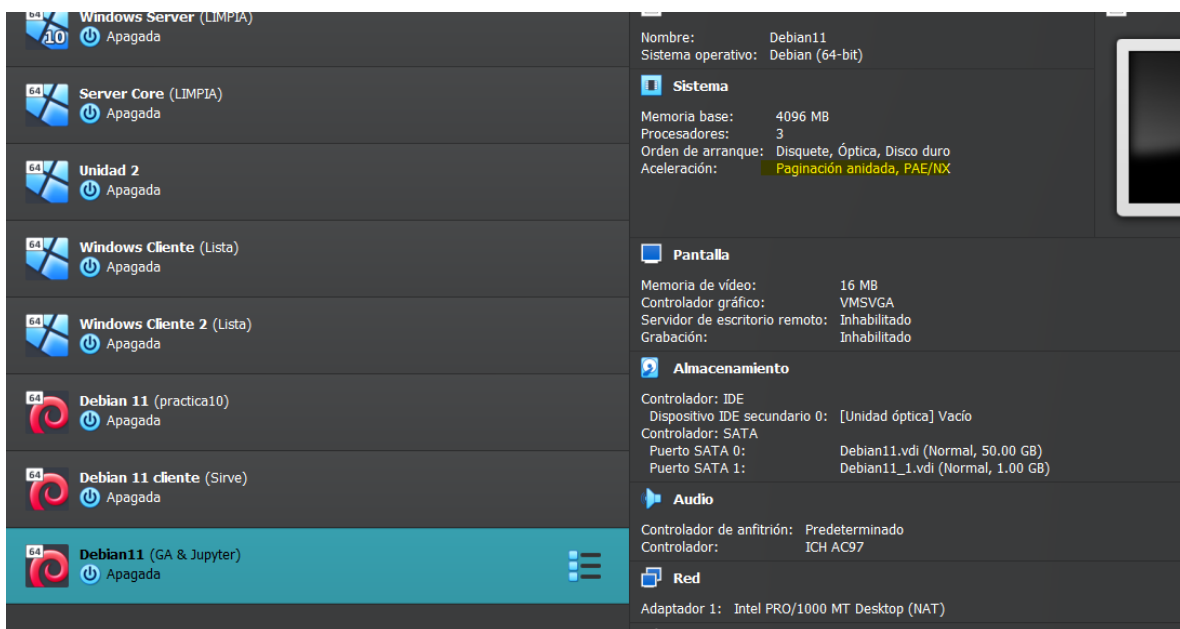
Nos dirigimos a las propiedades y en el apartado de sistema, en la pestaña de procesador seleccionamos habilitar PAE/NX.



Después en la pestaña Aceleración cambiamos la opción a Heredada y habilitar la paginación



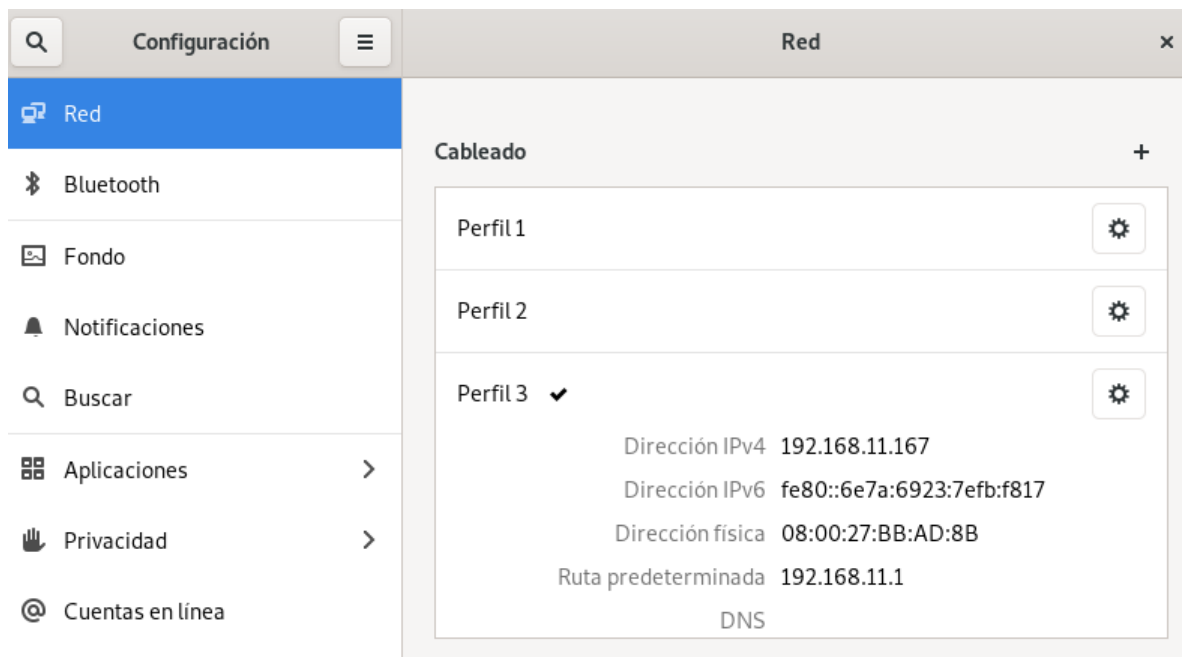
Debe aparecer la maquina con estas propiedades en el sistema.



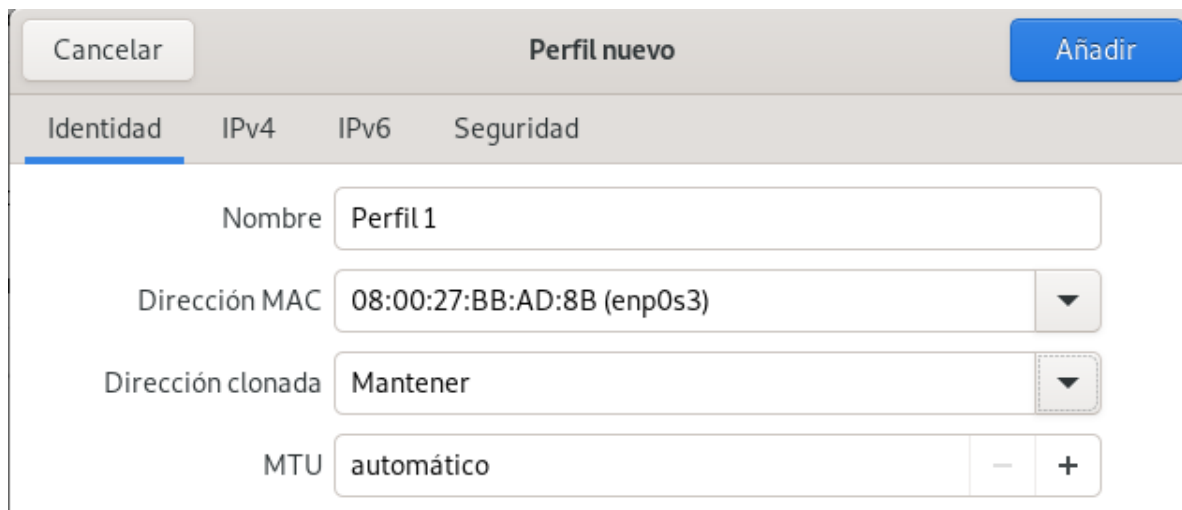
Instalaremos el servicio DHCP, usando apt install isc-dhcp-server, usando previamente su – root e ingresando la contraseña del usuario root. Es importante usar **SU – ROOT** cada que abrimos una terminal.

```
root@Debian:~# apt install isc-dhcp-server
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  libirs-export161 libiscfg-export163 policycoreutils selinux-utils
Paquetes sugeridos:
  isc-dhcp-server-ldap
Se instalarán los siguientes paquetes NUEVOS:
  isc-dhcp-server libirs-export161 libiscfg-export163 policycoreutils selinux-utils
0 actualizados, 5 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 1 703 kB de archivos.
Se utilizarán 6 915 kB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] s
Des:1 http://deb.debian.org/debian bullseye/main amd64 libiscfg-export163 amd64 1:9.11
.19+dfsg-2.1 [272 kB]
Des:2 http://deb.debian.org/debian bullseye/main amd64 libirs-export161 amd64 1:9.11.19
+dfsg-2.1 [245 kB]
Des:3 http://deb.debian.org/debian bullseye/main amd64 isc-dhcp-server amd64 4.4.1-2.3+
deb11u2 [554 kB]
Des:4 http://deb.debian.org/debian bullseye/main amd64 selinux-utils amd64 3.1-3 [142 k
B]
Des:5 http://deb.debian.org/debian bullseye/main amd64 policycoreutils amd64 3.1-3 [491
kB]
Descargados 1 703 kB en 1s (1 195 kB/s)
Preconfigurando paquetes ...
Seleccionando el paquete libiscfg-export163 previamente no seleccionado.
(Leyendo la base de datos ... 216110 ficheros o directorios instalados actualmente.)
Resolviendo dependencias...
libiscfg-export163 1:9.11.19+dfsg-2.1 amd64 deb
```

Mientras se instala el DHCP, nos dirigimos a configuración y en red añadiremos un perfil de cableado, en el cual se configurará una IP estática. Para el correcto funcionamiento y buenas practicas se debe usar una IP estática para el servidor. En este caso se usará 192.168.11.167, mascarará de subred 255.255.255.0 y default Gateway 192.168.11.1



Seleccionamos la MAC, mantenemos la dirección clonada y dejamos MTU en automático.



En direcciones asignamos la dirección IP, mascara de subred y default Gateway.

Cancelar Perfil nuevo Añadir

Identidad IPv4 IPv6 Seguridad

☒ Manual ☐ Desactivar
☐ Compartida con otros equipos

Direcciones

Dirección Máscara de red Puerta de enlace

DNS Automático ☒

Una vez que se instale el DHCP, probamos iniciar el servicio con `systemctl start isc-dhcp-server`. Es normal que nos salga este error, puesto que no hemos configurado el rango de direcciones y el tiempo que se ceden las direcciones IP

```

root@Debian:~# systemctl start isc-dhcp-server
Job for isc-dhcp-server.service failed because the control process exited with error code.
See "systemctl status isc-dhcp-server.service" and "journalctl -xe" for details.
root@Debian:~#

```

El comando `"journalctl -xe"` se utiliza para ver los registros del sistema, incluyendo mensajes de registro del sistema (logs) y mensajes del demonio del sistema, en tiempo real. Esta herramienta está disponible en sistemas operativos basados en systemd, como la mayoría de las distribuciones modernas de Linux.

Aquí hay una explicación de los elementos del comando:

- `"journalctl"`: Es el nombre del comando que se utiliza para acceder a los registros del sistema almacenados por el servicio journal de systemd.
- `"-xe"`: Son opciones que se utilizan para mostrar mensajes desde el final del registro (logs) y para mostrar mensajes anteriores a la interrupción actual del comando.

Al ejecutar `"journalctl -xe"`, la terminal mostrará mensajes de registro en tiempo real, y cuando llegues al final de los registros disponibles, te permitirá ver los mensajes anteriores a la interrupción del comando.

Esta herramienta puede ser útil para diagnosticar problemas del sistema, verificar errores o analizar eventos recientes que puedan estar afectando el rendimiento o el comportamiento del sistema. Además, también puede mostrar mensajes relacionados con el arranque del sistema, información de los servicios en ejecución y otros eventos importantes.

Buscamos algún error relacionado al servicio DHCP y en letras rojas nos señala que no se ha hecho la configuración del rango de direcciones en el archivo **dhcpd.conf** y no se asignado una interfaz

```
agus@Debian: ~
The job identifier is 3011.
jul 19 15:06:06 Debian isc-dhcp-server[3950]: Launching both IPv4 and IPv6 servers (pl>
jul 19 15:06:06 Debian dhcpd[3966]: Wrote 0 leases to leases file.
jul 19 15:06:06 Debian dhcpd[3966]:
jul 19 15:06:06 Debian dhcpd[3966]: No subnet declaration for enp0s3 (192.168.1.10).
jul 19 15:06:06 Debian dhcpd[3966]: ** Ignoring requests on enp0s3. If this is not wh>
jul 19 15:06:06 Debian dhcpd[3966]: you want, please write a subnet declaration
jul 19 15:06:06 Debian dhcpd[3966]: in your dhcpd.conf file for the network segment
jul 19 15:06:06 Debian dhcpd[3966]: to which interface enp0s3 is attached. **
jul 19 15:06:06 Debian dhcpd[3966]:
jul 19 15:06:06 Debian dhcpd[3966]: Not configured to listen on any interfaces!
jul 19 15:06:06 Debian dhcpd[3966]:
jul 19 15:06:06 Debian dhcpd[3966]: If you think you have received this message due to>
jul 19 15:06:06 Debian dhcpd[3966]: than a configuration issue please read the section>
jul 19 15:06:06 Debian dhcpd[3966]: bugs on either our web page at www.isc.org or in t>
jul 19 15:06:06 Debian dhcpd[3966]: before submitting a bug. These pages explain the>
jul 19 15:06:06 Debian dhcpd[3966]: process and the information we find helpful for de>
jul 19 15:06:06 Debian dhcpd[3966]: exiting.
jul 19 15:06:08 Debian isc-dhcp-server[3950]: Starting ISC DHCPv4 server: dhcpdcheck s>
jul 19 15:06:08 Debian isc-dhcp-server[3971]: failed!
jul 19 15:06:08 Debian isc-dhcp-server[3972]: failed!
jul 19 15:06:08 Debian systemd[1]: isc-dhcp-server.service: Control process exited, co>
Subject: Unit process exited
Defined-By: systemd
lines 2429-2455/2525 98%
```

Nos dirigimos al archivo ubicado en `/etc/default/isc-dhcp-server`. Y en la variable `INTERFACESv4` asignamos el nombre del adaptador de red que usa nuestro sistema. Normalmente es `enp0s3` pero para verificar esto usamos este el comando `ip add`

```
agus@Debian: ~
I /etc/default/isc-dhcp-server (Modified) Row 17 Col 21
# Defaults for isc-dhcp-server (sourced by /etc/init.d/isc-dhcp-server)

# Path to dhcpd's config file (default: /etc/dhcp/dhcpd.conf).
#DHCPDv4_CONF=/etc/dhcp/dhcpd.conf
#DHCPDv6_CONF=/etc/dhcp/dhcpd6.conf

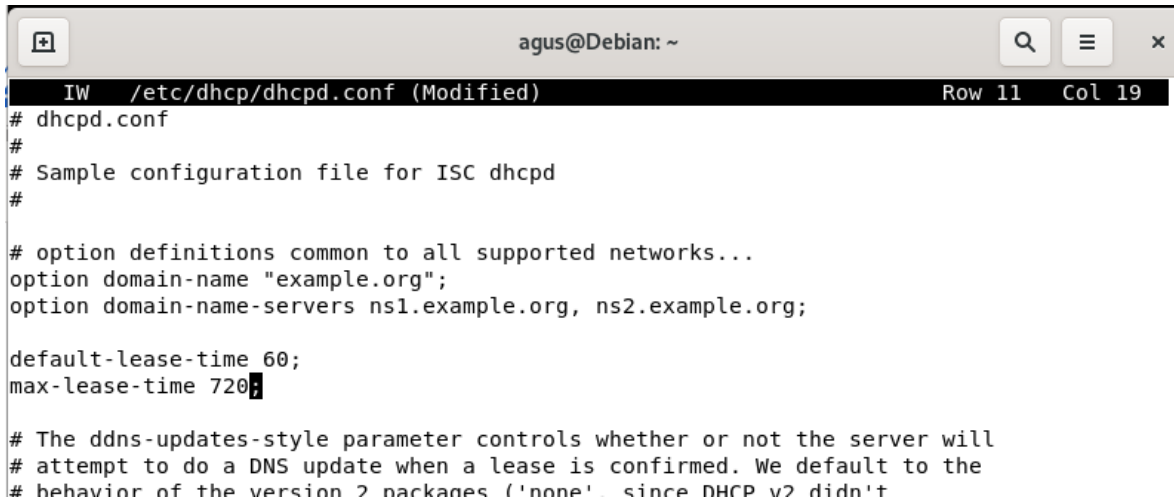
# Path to dhcpd's PID file (default: /var/run/dhcpd.pid).
#DHCPDv4_PID=/var/run/dhcpd.pid
#DHCPDv6_PID=/var/run/dhcpd6.pid

# Additional options to start dhcpd with.
# Don't use options -cf or -pf here; use DHCPD_CONF/ DHCPD_PID instead
#OPTIONS=""

# On what interfaces should the DHCP server (dhcpd) serve DHCP requests?
# Separate multiple interfaces with spaces, e.g. "eth0 eth1".
INTERFACESv4="enp0s3"
INTERFACESv6=""
```

Después nos dirigimos al archivo dhcpd.conf ubicado en /etc/dhcp/dhcpd.conf.

Primero modificamos el tiempo de actualización de dirección IP.

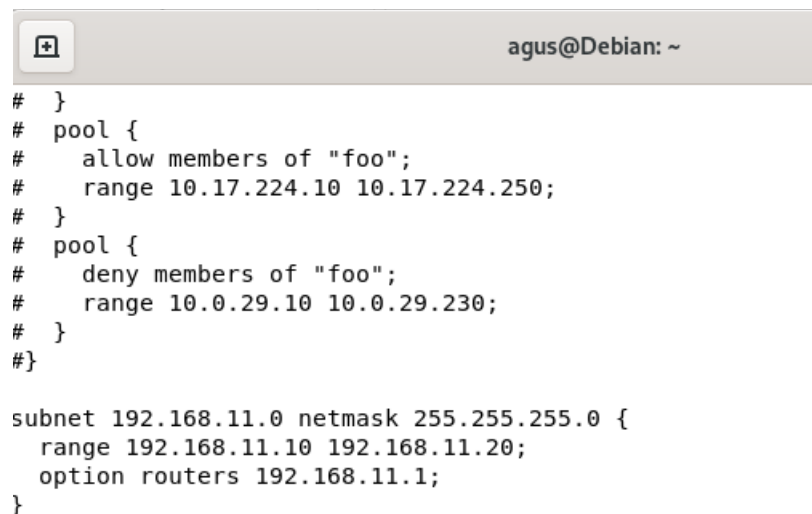


```
IW /etc/dhcp/dhcpd.conf (Modified) Row 11 Col 19
# dhcpd.conf
#
# Sample configuration file for ISC dhcpd
#
# option definitions common to all supported networks...
option domain-name "example.org";
option domain-name-servers ns1.example.org, ns2.example.org;

default-lease-time 60;
max-lease-time 720;

# The ddns-updates-style parameter controls whether or not the server will
# attempt to do a DNS update when a lease is confirmed. We default to the
# behavior of the version 2 packages ('none', since DHCP v2 didn't
```

Después al final del archivo escribiremos lo siguiente.



```
# }
# pool {
#   allow members of "foo";
#   range 10.17.224.10 10.17.224.250;
# }
# pool {
#   deny members of "foo";
#   range 10.0.29.10 10.0.29.230;
# }
#}

subnet 192.168.11.0 netmask 255.255.255.0 {
    range 192.168.11.10 192.168.11.20;
    option routers 192.168.11.1;
}
```

```
File /etc/dhcp/dhcpd.conf saved
root@Debian:~# █
```

Una vez hecho esto reiniciamos la máquina, después abrimos otra terminal y escribimos `systemctl status isc-dhcp-server`. El cual nos mostrara el estado del servicio DHCP.

Después de verificar que esté funcionando correctamente el servicio, iniciamos la máquina que tendrá la función de un cliente en el cual se otorgara una dirección IP a través de nuestro servidor.