

Configuración del sistema operativo, shell y software de soporte de red

Santiago Botero Garcia

Laura Natalia Perilla Quintero

Escuela Colombiana de Ingenieria Julio Garavito

AYSR-1L: Arquitectura y Servicios de Red

Ing. Jhon Alexander Pachon Pinzon

Septiembre 06, 2025

Resumen

Este informe presenta un enfoque integral para la configuración, administración y análisis de una infraestructura de tecnologías de la información (IT), combinando prácticas esenciales de virtualización, gestión de sistemas operativos y análisis de redes. A través de una serie de actividades prácticas y experimentos guiados, se busca desarrollar competencias técnicas clave que permitan comprender y operar en entornos IT modernos.

El proyecto contempla la instalación y configuración de sistemas operativos basados en Unix/Linux, así como el despliegue de máquinas virtuales mediante herramientas de virtualización. Además, se introduce el uso de herramientas especializadas como Packet Tracer y Wireshark para el diseño, simulación y análisis del tráfico de red, permitiendo una comprensión profunda de los protocolos de comunicación y la estructura de las redes.

Complementariamente, se abordan tareas de administración de sistemas mediante programación Shell, incluyendo la automatización de procesos, gestión de archivos y usuarios, y el uso del editor VI para la manipulación avanzada de texto. También se realiza un análisis detallado de tarjetas de red en distintos dispositivos, y se configura un entorno de compartición de archivos utilizando SMB/SAMBA en Solaris, con el objetivo de integrar sistemas operativos heterogéneos en una red funcional.

Las herramientas empleadas incluyen computadoras de laboratorio, acceso a Internet, software de virtualización, imágenes de sistemas operativos, Packet Tracer y Wireshark.

En conjunto, este informe proporciona una base práctica y conceptual para el desarrollo de habilidades en administración de redes y sistemas operativos, destacando la importancia de la

virtualización, el análisis de tráfico y la interoperabilidad entre plataformas en el contexto de infraestructuras IT actuales.

Palabras clave: Infraestructura IT, Administración de sistemas operativos, Programación Shell, Packet Tracer, Wireshark, Virtualización, Herramientas de redes, Compartición de archivos, Análisis de tráfico de red, SMB/SAMBA, Despliegue de máquinas virtuales, Sistemas Unix/Linux

Contenido

Resumen	2
Metodología	5
Experimentos	5
<i>Conociendo Packet Tracer</i>	6
<i>Rastreando Mensajes con Packet Tracer</i>	13
En la Red Real	24
<i>Usando Wireshark</i>	24
<i>Tarjetas de Red</i>	25
Software Base	30
<i>Programación en Shell – Unix</i>	30
<i>Editor VI en Linux/Unix</i>	32
<i>Despliegue de Máquinas Virtuales</i>	39
<i>Compartir Archivos</i>	42
Conociendo Cloud	47
Resultados	57
Experimentos	57
En Red Real	59
Software Base	63
Conclusiones	82
Referencias	84

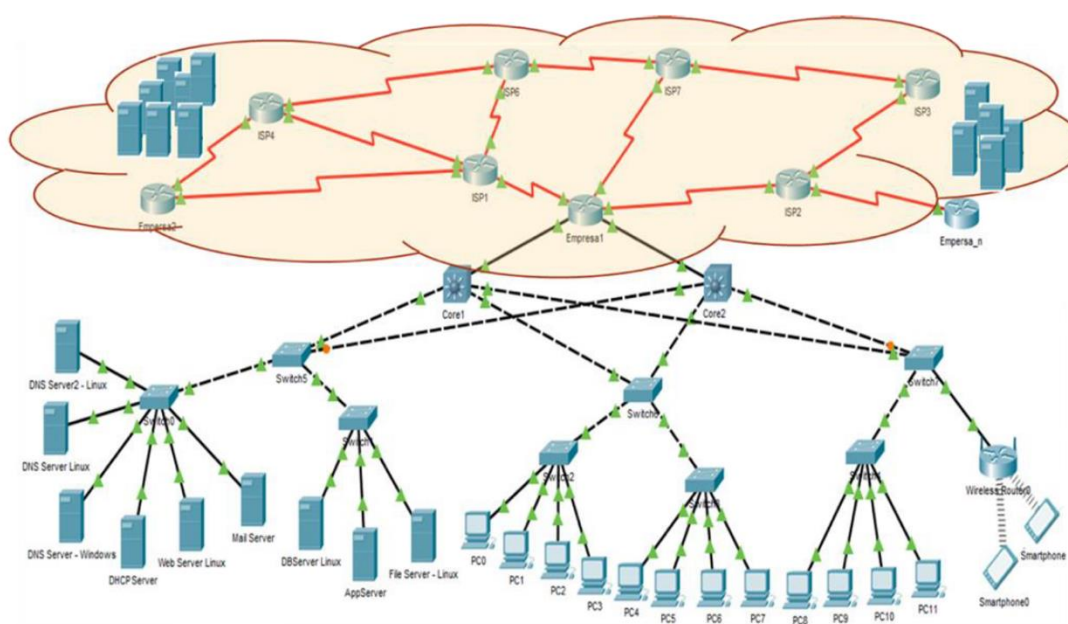
Metodología

Experimentos

En esta sección del informe, se abordarán una serie de actividades prácticas diseñadas para explorar y comprender la infraestructura tecnológica necesaria para el funcionamiento de redes y sistemas operativos.

Figura 1

Diagrama de una infraestructura tecnológica de red con servidores y dispositivos conectados.



Nota. Esta figura, proporcionada como parte del ejercicio de laboratorio, ilustra una infraestructura tecnológica de red compleja. En ella se evidencian los distintos servidores y dispositivos interconectados dentro de la red. Además, muestra cómo estos se conectan a través de switches, routers y conexiones WAN a varios proveedores de servicios de Internet (ISP). La imagen resalta la distribución de dispositivos finales como PCs y smartphones, que se conectan de manera local o a través de redes inalámbricas.

Con el objetivo de replicar la infraestructura tecnológica presentada en el diagrama anterior, es fundamental contar con computadoras y servidores que tengan un sistema operativo instalado, así como entender su funcionamiento desde la perspectiva del administrador de sistemas. Además, se pondrá énfasis en el uso de herramientas que faciliten la automatización de procesos administrativos.

A continuación, se detallan los experimentos que se llevarán a cabo, los cuales incluyen el uso de herramientas como Packet Tracer y la simulación de tráfico de red, lo cual permitirá a los estudiantes observar cómo se transmite la información en una red, analizar protocolos de red y crear diagramas que representen una infraestructura de red. Estas actividades son esenciales para comprender cómo interactúan los diferentes componentes de la red y el sistema operativo desde el punto de vista de administración y operación.

Conociendo Packet Tracer

Para ayudar en la comprensión de cómo funciona Cisco Packet Tracer, se ha creado un video explicativo que cubre los conceptos básicos del curso [Getting Started With Cisco Packet Tracer](#). Este curso tiene como objetivo guiar a los estudiantes a través de las herramientas y funciones fundamentales de la plataforma, permitiéndoles desarrollar habilidades en la simulación de redes.

El video elaborado acompaña este informe como recurso práctico y muestra el proceso de creación de una red sencilla, incluyendo la adición de dispositivos, el cambio de nombres, las conexiones físicas, y la configuración de los dispositivos finales, tanto por cable como de forma inalámbrica.

[Mira el Video: “Guía Completa de Cisco Packet Tracer: Instalación, Interfaz y Creación de una Red Doméstica”]

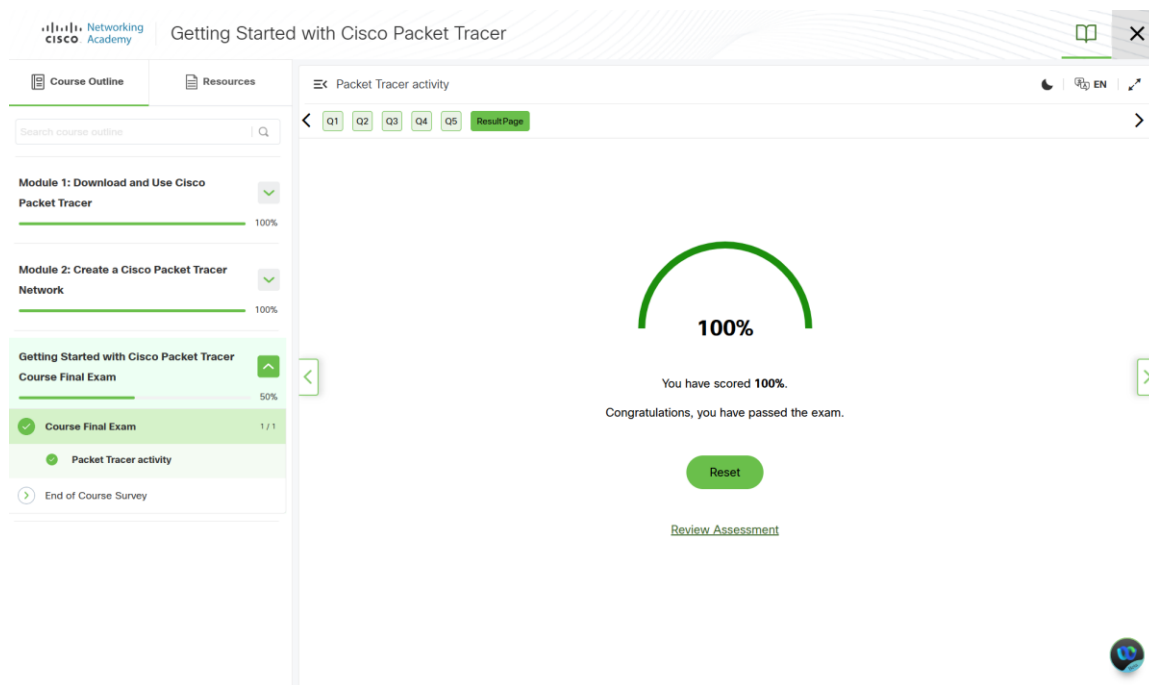
(<https://drive.google.com/file/d/1DOLoGQZ9RwVJIVIAcDtKlS3jK6Emc616/view?usp=sharing>) (Cisco Networking Academy, n.d.)

Como parte del trabajo práctico, todos los integrantes del grupo completaron dicho curso a través de la plataforma Cisco Networking Academy. Tal como se indicó anteriormente, este contenido cubre desde la instalación del software hasta la construcción y análisis de redes básicas.

A continuación, se presentan capturas de pantalla correspondientes a la finalización del curso por parte de uno de los integrantes.

Figura 2

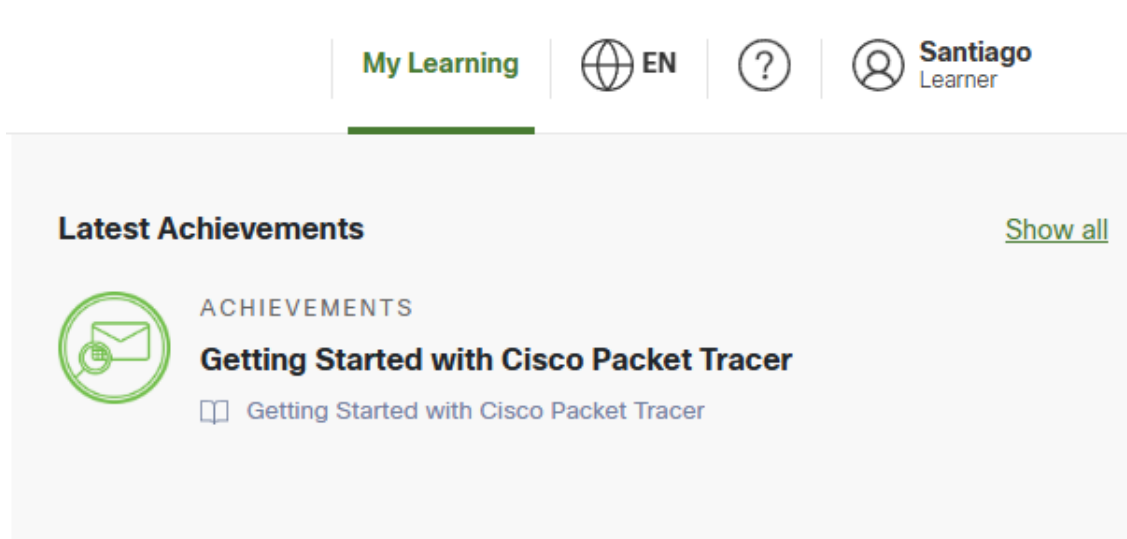
Puntaje final del curso



Nota. Captura del puntaje obtenido por Santiago en el examen final del curso, con una calificación del **100%**

Figura 3

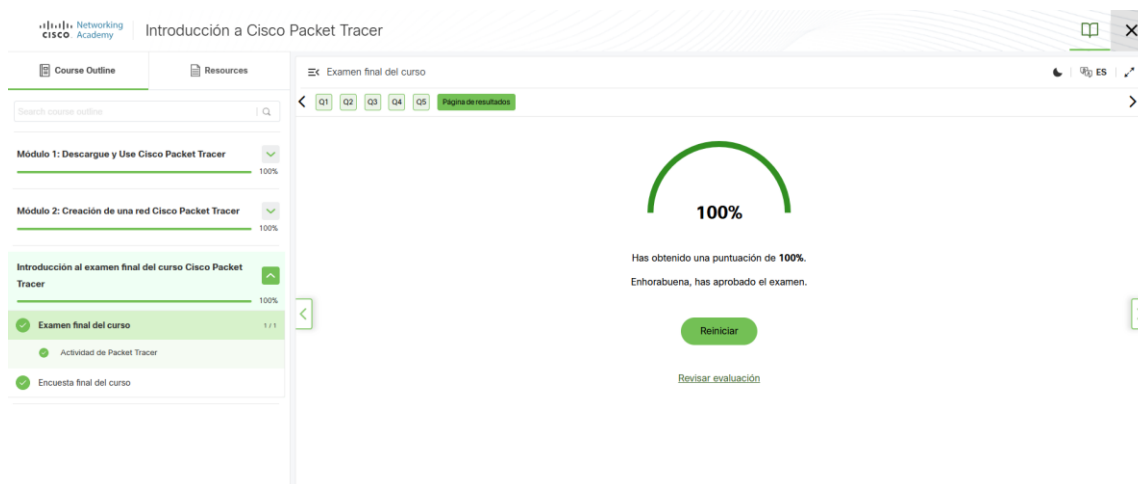
Logro obtenido en Cisco Networking Academy



Nota. Imagen que muestra el reconocimiento registrado en la sección "Latest Achievements", certificando la finalización exitosa del curso

Figura 4

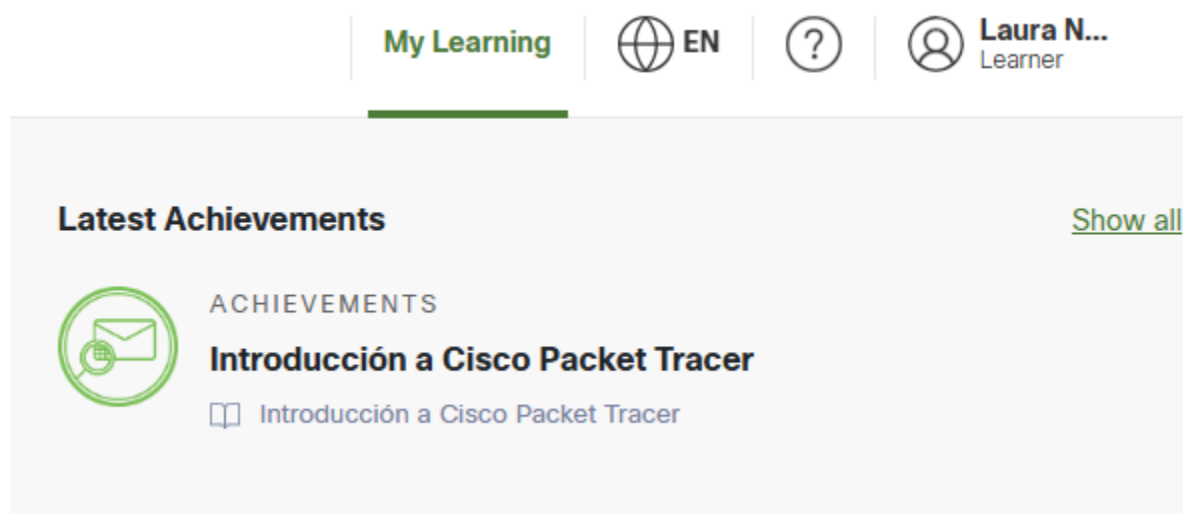
Puntaje final



Nota. Puntaje obtenido por Natalia en el examen final del curso, con una calificación del **100%**

Figura 5

Logro obtenido en Cisco Networking Academy

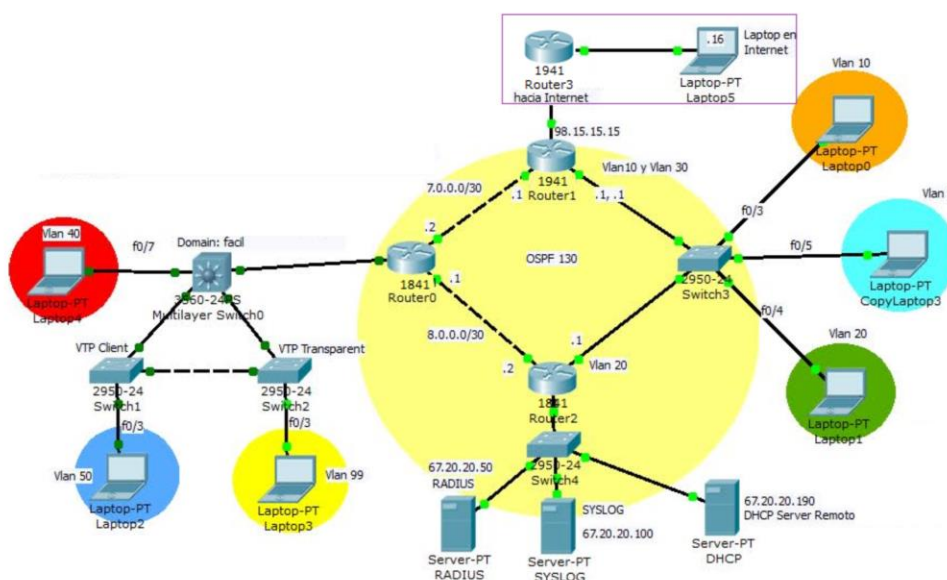


Nota. Imagen que muestra el reconocimiento registrado en la sección, lo cual certifica la finalización exitosa del curso

Después de la finalización del quiz y antes de proceder con la configuración de la red, se tomó como referencia una imagen incluida en el informe recibido previamente. Dicha imagen sirvió como base para el diseño y la topología de la implementación de OSPF realizada en Cisco Packet Tracer, adaptando y ajustando la estructura a las necesidades específicas del proyecto. A partir de este modelo, se realizaron las modificaciones necesarias para optimizar la conectividad, incluyendo el cambio de enlaces crossover a serial para simular una red WAN, la instalación del módulo HWIC-2T en el Router1 (modelo 1941) y la configuración de direcciones IP y bases de datos de VLAN.

Figura 6

Topología de red original utilizada como referencia para la implementación de OSPF



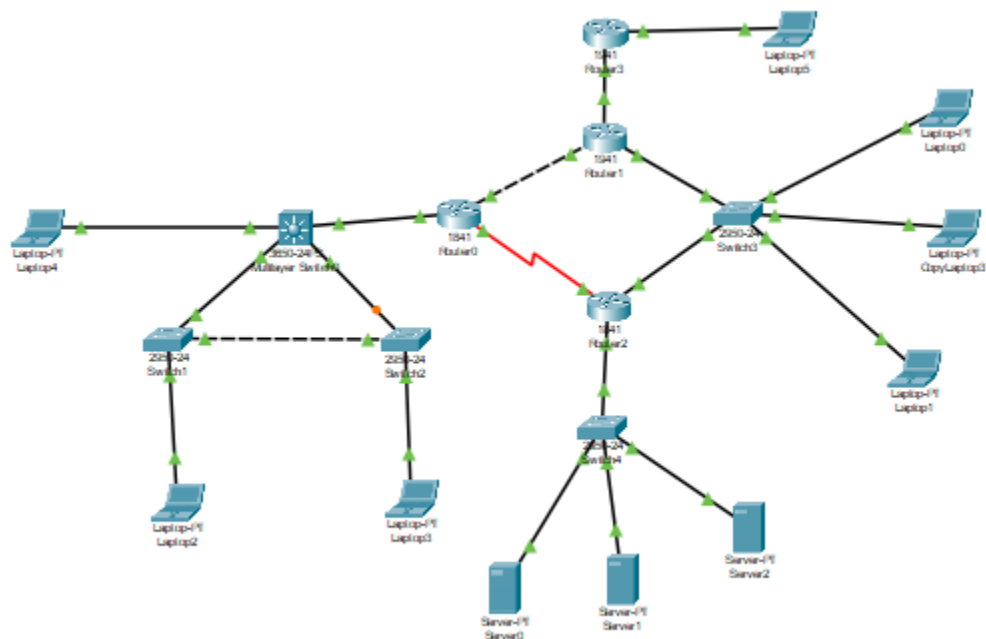
Nota. Imagen tomada del informe recibido, la cual sirvió como inspiración para el diseño de la red implementada en Cisco Packet Tracer. Esta topología base permitió definir la disposición de routers, switches, servidores y dispositivos finales, así como la segmentación en VLANs y la asignación de direcciones IP, sobre la cual se aplicaron las configuraciones de OSPF para lograr una red funcional y escalable.

Con base en el diseño de referencia mostrado anteriormente, se procedió a continuar con la implementación práctica de la red bajo el protocolo OSPF. Esta fase se llevó a cabo con el objetivo de poner en práctica los conocimientos adquiridos durante el curso y el quiz, así como para simular un entorno de red más cercano a un escenario real.

Para facilitar la revisión y replicación del trabajo, los archivos del proyecto en formato .pkz se encuentran adjuntos a este informe. En particular, el archivo correspondiente a la implementación realizada por Santiago se denomina `ospf-santiago-implementation.pkz`, mientras que el archivo correspondiente a la implementación realizada por Natalia se denomina `Red_Natalia.pkz`. Ambos contienen la configuración completa de la red, incluyendo las

Figura 8

Implementación de OSPF en Cisco Packet Tracer



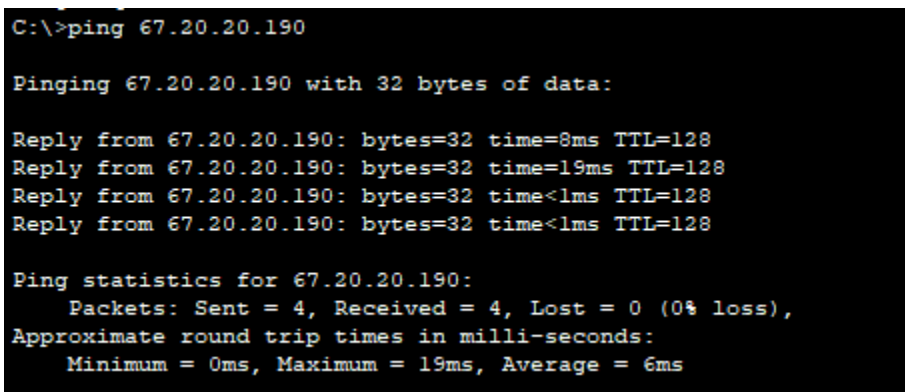
Nota. Esta figura muestra la implementación del protocolo OSPF (Open Shortest Path First) realizado por Natalia en un entorno simulado con Cisco Packet Tracer.

Rastreando Mensajes con Packet Tracer

Con la configuración inicial realizada por Santiago, se procedió a la siguiente fase del trabajo: la trazabilidad de paquetes. Para ello, se ejecutó una prueba de conectividad mediante un comando ping desde el servidor RADIUS hacia el servidor DHCP, obteniendo desde la consola un promedio de 6 ms de latencia.

Figura 9

Resultado del comando *ping* desde el servidor RADIUS hacia el servidor DHCP.



```
C:\>ping 67.20.20.190

Pinging 67.20.20.190 with 32 bytes of data:

Reply from 67.20.20.190: bytes=32 time=8ms TTL=128
Reply from 67.20.20.190: bytes=32 time=19ms TTL=128
Reply from 67.20.20.190: bytes=32 time<1ms TTL=128
Reply from 67.20.20.190: bytes=32 time<1ms TTL=128

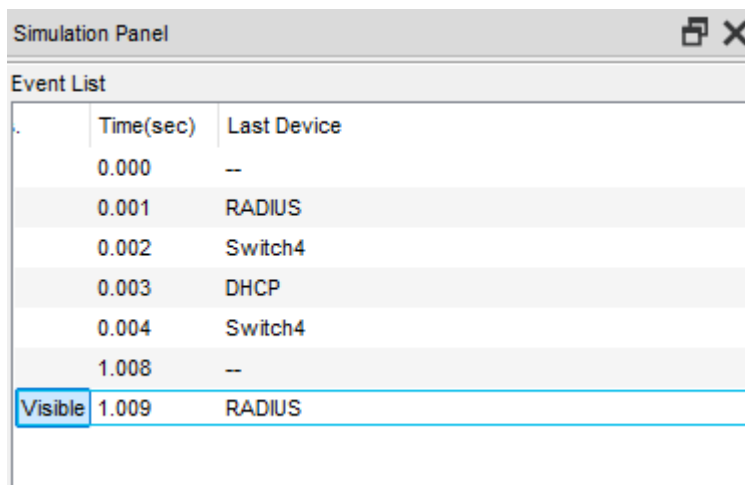
Ping statistics for 67.20.20.190:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 19ms, Average = 6ms
```

Nota. Captura de la consola que muestra el promedio de latencia obtenido (6 ms) durante la prueba de conectividad entre el servidor RADIUS y el servidor DHCP.

No obstante, utilizando el modo simulador de Cisco Packet Tracer es posible llevar un control más detallado de lo que ocurre en la transmisión de datos. Este modo permite visualizar el recorrido de cada paquete, su estado y el tiempo de tránsito, lo que facilita el análisis del comportamiento de la red.

Figura 10

Vista del *Simulation Mode* en Cisco Packet Tracer con el envío de cinco paquetes.



Simulation Panel	
Event List	
	Time(sec) Last Device
	0.000 --
	0.001 RADIUS
	0.002 Switch4
	0.003 DHCP
	0.004 Switch4
	1.008 --
Visible	1.009 RADIUS

Nota. Captura del modo simulador de Cisco Packet Tracer mostrando el envío y seguimiento de cinco paquetes, permitiendo observar en detalle el flujo de datos y su paso por los diferentes dispositivos de la red.

Durante el análisis en modo simulador de Cisco Packet Tracer, si se abre cada paquete enviado es posible observar el paso a paso de su recorrido a través de las distintas capas del modelo OSI (Open Systems Interconnection). Este modelo conceptual divide el proceso de comunicación en siete capas:

7. **Capa de Aplicación:** Es la única capa que interactúa directamente con el usuario. Su función principal es ofrecer servicios de red a las aplicaciones (como navegadores web o clientes de correo electrónico), aunque dichas aplicaciones no forman parte de esta capa. La capa de aplicación gestiona los protocolos que permiten la comunicación, como HTTP (para navegación web) o SMTP (para envío de correos electrónicos), y se encarga de presentar los datos al usuario de manera comprensible.
6. **Capa de Presentación:** Su función es preparar los datos para que la capa de aplicación pueda utilizarlos. Realiza tareas como:

- Traducción de formatos entre diferentes sistemas.
 - Cifrado y descifrado de datos para garantizar la seguridad.
 - Compresión de información para optimizar el rendimiento y reducir el volumen de datos transmitidos.
5. **Capa de Sesión:** Controla el inicio, mantenimiento y cierre de la comunicación entre dispositivos. Define lo que se conoce como una "sesión". Además, puede establecer puntos de control (checkpoints) durante una transferencia de datos extensa. En caso de interrupción, estos puntos permiten reanudar la transferencia desde el último punto registrado, evitando repetir todo el proceso.
4. **Capa de Transporte:** Responsable de la comunicación de extremo a extremo. Protocolos comunes en esta capa incluyen TCP y UDP. Divide los datos recibidos en segmentos y se encarga de:
- Reensamblar los datos al llegar al destino.
 - Aplicar control de flujo, ajustando la velocidad de envío para no saturar al receptor.
 - Realizar control de errores, verificando la integridad de los datos y solicitando reenvíos si es necesario.
3. **Capa de Red:** Encargada de la comunicación entre redes distintas. Divide los segmentos en paquetes y determina la mejor ruta física (routing) para que los datos lleguen a su destino. Algunos de los protocolos más importantes en esta capa son IP, ICMP, IGMP y IPsec.
2. **Capa de Enlace de Datos:** Opera dentro de una misma red. Toma los paquetes de la capa de red y los convierte en tramas. Además, proporciona control de flujo y de errores en redes locales.

1. **Capa Física:** Es la base del modelo OSI y abarca el hardware que transmite los datos físicamente, como cables, conectores, switches, y señales eléctricas o ópticas. Aquí, los datos se convierten en un flujo de bits (1s y 0s) para ser transmitidos por los medios físicos. Esta capa también se asegura de que el emisor y receptor usen el mismo tipo de señal para interpretar correctamente los datos.

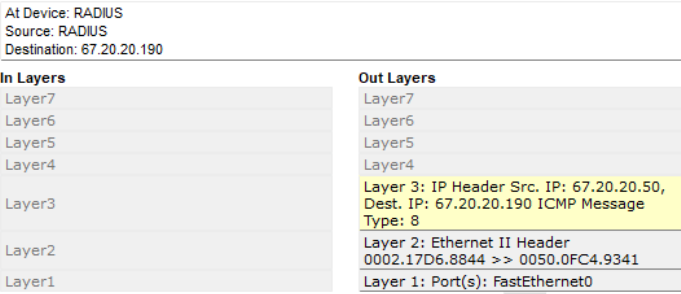
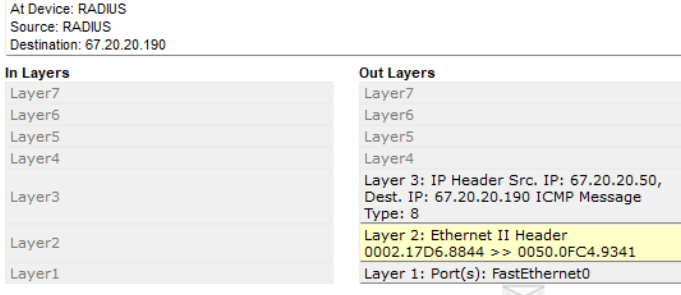
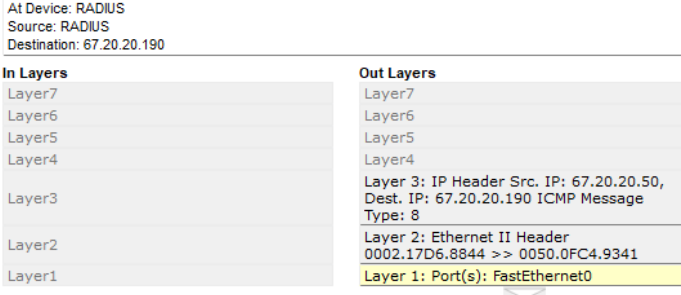
(What Is The OSI Model?, s. f.)

En Packet Tracer, al inspeccionar un paquete, se puede ver cómo la información se encapsula y desencapsula en cada capa, lo que permite comprender de forma visual y detallada el flujo de datos.

En este caso, para documentar de forma clara y ordenada el análisis, se optó por representar el recorrido del paquete mediante una línea de tiempo en tabla, en la que cada fila corresponde a un paso específico dentro del proceso de encapsulación y desencapsulación según el modelo OSI.

Esta estructura permite asociar cada captura de pantalla con la capa correspondiente, ofreciendo una descripción breve de lo que ocurre en ese momento del tránsito del paquete.

De esta manera, el lector puede seguir de forma secuencial cómo la información se transforma y avanza a través de las distintas capas, desde la Aplicación hasta la Física, facilitando la comprensión del flujo de datos y el papel que desempeña cada capa en la comunicación de red.

Paso / Capa del modelo OSI	Evidencia visual (captura de pantalla)
<p>En el dispositivo RADIUS, el proceso de ping inicia la generación de una solicitud ICMP Echo (tipo 8) con destino a la dirección IP 67.20.20.190. Como el paquete no especifica una dirección IP de origen inicialmente, el dispositivo asigna automáticamente su propia IP (67.20.20.50) como origen. Luego, verifica que el destino esté en la misma subred y procede a determinar el siguiente salto. Finalmente, el paquete es encapsulado en una trama Ethernet que incluye los encabezados correspondientes antes de ser enviado a la capa de enlace.</p>	 <p>At Device: RADIUS Source: RADIUS Destination: 67.20.20.190</p> <p>In Layers</p> <ul style="list-style-type: none"> Layer7 Layer6 Layer5 Layer4 Layer3 Layer2 Layer1 <p>Out Layers</p> <ul style="list-style-type: none"> Layer7 Layer6 Layer5 Layer4 Layer 3: IP Header Src. IP: 67.20.20.50, Dest. IP: 67.20.20.190 ICMP Message Type: 8 Layer 2: Ethernet II Header 0002.17D6.8844 >> 0050.0FC4.9341 Layer 1: Port(s): FastEthernet0 <p>1. The Ping process starts the next ping request. 2. The Ping process creates an ICMP Echo Request message and sends it to the lower process. 3. The source IP address is not specified. The device sets it to the port's IP address. 4. The destination IP address is in the same subnet. The device sets the next-hop to destination.</p>
<p>RADIUS identifica que la dirección IP de destino es unicast y consulta su tabla ARP para obtener la dirección MAC correspondiente. Al encontrar una entrada válida, asigna dicha MAC como la dirección de destino en el encabezado Ethernet. La dirección MAC de origen también es establecida y el paquete se encapsula completamente en una trama Ethernet, lista para ser transmitida a través del medio físico. Este paso garantiza que el paquete pueda ser entregado correctamente a través de la red local.</p>	 <p>At Device: RADIUS Source: RADIUS Destination: 67.20.20.190</p> <p>In Layers</p> <ul style="list-style-type: none"> Layer7 Layer6 Layer5 Layer4 Layer3 Layer2 Layer1 <p>Out Layers</p> <ul style="list-style-type: none"> Layer7 Layer6 Layer5 Layer4 Layer 3: IP Header Src. IP: 67.20.20.50, Dest. IP: 67.20.20.190 ICMP Message Type: 8 Layer 2: Ethernet II Header 0002.17D6.8844 >> 0050.0FC4.9341 Layer 1: Port(s): FastEthernet0 <p>1. The next-hop IP address is a unicast. The ARP process looks it up in the ARP table. 2. The next-hop IP address is in the ARP table. The ARP process sets the frame's destination MAC address to the one found in the table. 3. The device encapsulates the PDU into an Ethernet frame.</p>
<p>El dispositivo RADIUS, a través de su interfaz FastEthernet0, transmite la trama Ethernet al destino. En este punto, el paquete ya contiene toda la información necesaria a nivel de capa de red y de enlace, por lo que simplemente es enviado al siguiente nodo en la red, continuando su camino hacia la dirección IP 67.20.20.190 como parte del proceso de ping iniciado previamente.</p>	 <p>At Device: RADIUS Source: RADIUS Destination: 67.20.20.190</p> <p>In Layers</p> <ul style="list-style-type: none"> Layer7 Layer6 Layer5 Layer4 Layer3 Layer2 Layer1 <p>Out Layers</p> <ul style="list-style-type: none"> Layer7 Layer6 Layer5 Layer4 Layer 3: IP Header Src. IP: 67.20.20.50, Dest. IP: 67.20.20.190 ICMP Message Type: 8 Layer 2: Ethernet II Header 0002.17D6.8844 >> 0050.0FC4.9341 Layer 1: Port(s): FastEthernet0 <p>1. FastEthernet0 sends out the frame.</p>

<p>El switch4 recibe la trama Ethernet a través de su puerto FastEthernet0/2. En esta etapa, el switch simplemente detecta la llegada del paquete, que contiene una dirección MAC de origen (0002.17D6.8844) y una de destino (0050.0FC4.9341), ambas visibles en la cabecera de capa 2. Aún no se toma ninguna decisión de reenvío, solo se registra la recepción del paquete.</p>	<p>At Device: Switch4 Source: RADIUS Destination: 67.20.20.190</p> <div> <div> In Layers Layer7 Layer6 Layer5 Layer4 Layer3 Layer 2: Ethernet II Header 0002.17D6.8844 >> 0050.0FC4.9341 Layer 1: Port FastEthernet0/2 </div> <div> Out Layers Layer7 Layer6 Layer5 Layer4 Layer3 Layer 2: Ethernet II Header 0002.17D6.8844 >> 0050.0FC4.9341 Layer 1: Port(s): FastEthernet0/4 </div> </div> <p>1. FastEthernet0/2 receives the frame.</p>
<p>Al recibir la trama, el switch analiza su tabla MAC y confirma que la dirección MAC de origen ya está registrada en ella. Al tratarse de una trama unicast, el switch consulta su tabla para determinar el puerto de salida correspondiente a la dirección MAC de destino. Gracias a esta verificación, el switch sabe hacia qué puerto debe enviar la trama para alcanzar su destino.</p>	<p>At Device: Switch4 Source: RADIUS Destination: 67.20.20.190</p> <div> <div> In Layers Layer7 Layer6 Layer5 Layer4 Layer3 Layer 2: Ethernet II Header 0002.17D6.8844 >> 0050.0FC4.9341 Layer 1: Port FastEthernet0/2 </div> <div> Out Layers Layer7 Layer6 Layer5 Layer4 Layer3 Layer 2: Ethernet II Header 0002.17D6.8844 >> 0050.0FC4.9341 Layer 1: Port(s): FastEthernet0/4 </div> </div> <p>1. The frame source MAC address was found in the MAC table of Switch. 2. This is a unicast frame. Switch looks in its MAC table for the destination MAC address.</p>
<p>Después de identificar el puerto de salida como FastEthernet0/4, el switch verifica que este puerto es parte de un enlace troncal (trunk) y que el número de VLAN correspondiente al paquete está permitido en ese enlace. Como se cumplen ambas condiciones, el switch está autorizado a reenviar la trama por ese puerto sin necesidad de modificar la encapsulación VLAN.</p>	<p>At Device: Switch4 Source: RADIUS Destination: 67.20.20.190</p> <div> <div> In Layers Layer7 Layer6 Layer5 Layer4 Layer3 Layer 2: Ethernet II Header 0002.17D6.8844 >> 0050.0FC4.9341 Layer 1: Port FastEthernet0/2 </div> <div> Out Layers Layer7 Layer6 Layer5 Layer4 Layer3 Layer 2: Ethernet II Header 0002.17D6.8844 >> 0050.0FC4.9341 Layer 1: Port(s): FastEthernet0/4 </div> </div> <p>1. The outgoing port is a trunk port and the incoming port VLAN number is allowed in the trunk. Switch sends out the frame to that port.</p>
<p>Finalmente, el switch4 reenvía la trama a través de su puerto FastEthernet0/4. Esta acción marca la salida física del paquete desde el switch hacia su siguiente destino en la red, manteniendo intactos los encabezados Ethernet ya que no se requiere modificación adicional.</p>	<p>At Device: Switch4 Source: RADIUS Destination: 67.20.20.190</p> <div> <div> In Layers Layer7 Layer6 Layer5 Layer4 Layer3 Layer 2: Ethernet II Header 0002.17D6.8844 >> 0050.0FC4.9341 Layer 1: Port FastEthernet0/2 </div> <div> Out Layers Layer7 Layer6 Layer5 Layer4 Layer3 Layer 2: Ethernet II Header 0002.17D6.8844 >> 0050.0FC4.9341 Layer 1: Port(s): FastEthernet0/4 </div> </div> <p>1. FastEthernet0/4 sends out the frame.</p>

<p>El dispositivo DHCP recibe la trama Ethernet a través de su interfaz FastEthernet0. En esta trama, se encapsula un mensaje ICMP Echo Request proveniente del dispositivo RADIUS, con dirección IP de origen 67.20.20.50 y dirección de destino 67.20.20.190. El encabezado de capa 2 muestra la dirección MAC de origen 0002.17D6.8844 y de destino 0050.0FC4.9341, coincidiendo con la interfaz de DHCP, por lo que el paquete es aceptado para su procesamiento.</p>	<div> At Device: DHCP Source: RADIUS Destination: 67.20.20.190 </div> <div> <div> In Layers Layer7 Layer6 Layer5 Layer4 Layer 3: IP Header Src. IP: 67.20.20.50, Dest. IP: 67.20.20.190 ICMP Message Type: 8 Layer 2: Ethernet II Header 0002.17D6.8844 >> 0050.0FC4.9341 Layer 1: Port FastEthernet0 </div> <div> Out Layers Layer7 Layer6 Layer5 Layer4 Layer 3: IP Header Src. IP: 67.20.20.190, Dest. IP: 67.20.20.50 ICMP Message Type: 0 Layer 2: Ethernet II Header 0050.0FC4.9341 >> 0002.17D6.8844 Layer 1: Port(s): FastEthernet0 </div> </div> <div> 1. FastEthernet0 receives the frame. </div>
<p>El dispositivo DHCP verifica que la dirección MAC de destino en el encabezado Ethernet coincide con la dirección de su interfaz de red. Esto permite que el paquete sea procesado más allá de la capa 2. Posteriormente, el dispositivo desencapsula la PDU de la trama Ethernet, extrayendo los datos de la capa de red, donde se encuentra el paquete ICMP.</p>	<div> At Device: DHCP Source: RADIUS Destination: 67.20.20.190 </div> <div> <div> In Layers Layer7 Layer6 Layer5 Layer4 Layer 3: IP Header Src. IP: 67.20.20.50, Dest. IP: 67.20.20.190 ICMP Message Type: 8 Layer 2: Ethernet II Header 0002.17D6.8844 >> 0050.0FC4.9341 Layer 1: Port FastEthernet0 </div> <div> Out Layers Layer7 Layer6 Layer5 Layer4 Layer 3: IP Header Src. IP: 67.20.20.190, Dest. IP: 67.20.20.50 ICMP Message Type: 0 Layer 2: Ethernet II Header 0050.0FC4.9341 >> 0002.17D6.8844 Layer 1: Port(s): FastEthernet0 </div> </div> <div> 1. The frame's destination MAC address matches the receiving port's MAC address, the broadcast address, or a multicast address. 2. The device decapsulates the PDU from the Ethernet frame. </div>
<p>El paquete IP es procesado por DHCP ya que su dirección IP de destino (67.20.20.190) coincide con la IP del dispositivo. Al identificar que se trata de un mensaje ICMP tipo 8 (Echo Request), el dispositivo entrega el paquete al proceso ICMP interno para su análisis y gestión. DHCP se prepara así para generar una respuesta al mensaje recibido.</p>	<div> At Device: DHCP Source: RADIUS Destination: 67.20.20.190 </div> <div> <div> In Layers Layer7 Layer6 Layer5 Layer4 Layer 3: IP Header Src. IP: 67.20.20.50, Dest. IP: 67.20.20.190 ICMP Message Type: 8 Layer 2: Ethernet II Header 0002.17D6.8844 >> 0050.0FC4.9341 Layer 1: Port FastEthernet0 </div> <div> Out Layers Layer7 Layer6 Layer5 Layer4 Layer 3: IP Header Src. IP: 67.20.20.190, Dest. IP: 67.20.20.50 ICMP Message Type: 0 Layer 2: Ethernet II Header 0050.0FC4.9341 >> 0002.17D6.8844 Layer 1: Port(s): FastEthernet0 </div> </div> <div> 1. The packet's destination IP address matches the device's IP address or the broadcast address. The device de-encapsulates the packet. 2. The packet is an ICMP packet. The ICMP process processes it. 3. The ICMP process received an Echo Request message. </div>

<p>Como respuesta al mensaje ICMP Echo Request recibido, el dispositivo DHCP genera un mensaje ICMP tipo 0 (Echo Reply). Este nuevo paquete conserva las direcciones IP del mensaje original, pero invierte los roles de origen y destino. La respuesta se encapsula nuevamente con dirección MAC de origen 0050.0FC4.9341 (de DHCP) y destino 0002.17D6.8844 (de RADIUS), preparándose para su envío.</p>	<div> At Device: DHCP Source: RADIUS Destination: 67.20.20.190 </div> <div> <div> In Layers Layer7 Layer6 Layer5 Layer4 Layer 3: IP Header Src. IP: 67.20.20.50, Dest. IP: 67.20.20.190 ICMP Message Type: 8 Layer 2: Ethernet II Header 0002.17D6.8844 >> 0050.0FC4.9341 Layer 1: Port FastEthernet0 </div> <div> Out Layers Layer7 Layer6 Layer5 Layer4 Layer 3: IP Header Src. IP: 67.20.20.190, Dest. IP: 67.20.20.50 ICMP Message Type: 0 Layer 2: Ethernet II Header 0050.0FC4.9341 >> 0002.17D6.8844 Layer 1: Port(s): FastEthernet0 </div> </div> <div> 1. The ICMP process replies to the Echo Request by setting ICMP type to Echo Reply. 2. The ICMP process sends an Echo Reply. 3. The destination IP address is in the same subnet. The device sets the next-hop to destination. </div>
<p>El dispositivo DHCP consulta su tabla ARP al detectar que la dirección IP de siguiente salto (la del dispositivo RADIUS) es unicast. Al encontrar una entrada válida, el dispositivo establece la dirección MAC de destino en el encabezado Ethernet. Finalmente, encapsula el mensaje ICMP Echo Reply en una nueva trama Ethernet lista para su transmisión.</p>	<div> At Device: DHCP Source: RADIUS Destination: 67.20.20.190 </div> <div> <div> In Layers Layer7 Layer6 Layer5 Layer4 Layer 3: IP Header Src. IP: 67.20.20.50, Dest. IP: 67.20.20.190 ICMP Message Type: 8 Layer 2: Ethernet II Header 0002.17D6.8844 >> 0050.0FC4.9341 Layer 1: Port FastEthernet0 </div> <div> Out Layers Layer7 Layer6 Layer5 Layer4 Layer 3: IP Header Src. IP: 67.20.20.190, Dest. IP: 67.20.20.50 ICMP Message Type: 0 Layer 2: Ethernet II Header 0050.0FC4.9341 >> 0002.17D6.8844 Layer 1: Port(s): FastEthernet0 </div> </div> <div> 1. The next-hop IP address is a unicast. The ARP process looks it up in the ARP table. 2. The next-hop IP address is in the ARP table. The ARP process sets the frame's destination MAC address to the one found in the table. 3. The device encapsulates the PDU into an Ethernet frame. </div>
<p>DHCP transmite la trama de respuesta ICMP a través de su interfaz FastEthernet0. Esta acción marca el envío del mensaje Echo Reply de vuelta hacia el origen del ping, RADIUS, completando así la mitad del proceso ICMP, a la espera de que el mensaje sea recibido y reconocido por el dispositivo que originó la solicitud.</p>	<div> At Device: DHCP Source: RADIUS Destination: 67.20.20.190 </div> <div> <div> In Layers Layer7 Layer6 Layer5 Layer4 Layer 3: IP Header Src. IP: 67.20.20.50, Dest. IP: 67.20.20.190 ICMP Message Type: 8 Layer 2: Ethernet II Header 0002.17D6.8844 >> 0050.0FC4.9341 Layer 1: Port FastEthernet0 </div> <div> Out Layers Layer7 Layer6 Layer5 Layer4 Layer 3: IP Header Src. IP: 67.20.20.190, Dest. IP: 67.20.20.50 ICMP Message Type: 0 Layer 2: Ethernet II Header 0050.0FC4.9341 >> 0002.17D6.8844 Layer 1: Port(s): FastEthernet0 </div> </div> <div> 1. FastEthernet0 sends out the frame. </div>

<p>El Switch4 recibe la trama por el puerto FastEthernet0/4. Esta trama contiene una cabecera Ethernet con la dirección MAC de origen 0050.0FC4.9341 y la dirección MAC de destino 0002.17D6.8844. El dispositivo ahora se prepara para procesar esta información y reenviarla al destino correspondiente según su tabla MAC.</p>	<p>At Device: Switch4 Source: RADIUS Destination: 67.20.20.190</p> <table border="1"> <thead> <tr> <th>In Layers</th> <th>Out Layers</th> </tr> </thead> <tbody> <tr><td>Layer7</td><td>Layer7</td></tr> <tr><td>Layer6</td><td>Layer6</td></tr> <tr><td>Layer5</td><td>Layer5</td></tr> <tr><td>Layer4</td><td>Layer4</td></tr> <tr><td>Layer3</td><td>Layer3</td></tr> <tr> <td>Layer 2: Ethernet II Header 0050.0FC4.9341 >> 0002.17D6.8844</td> <td>Layer 2: Ethernet II Header 0050.0FC4.9341 >> 0002.17D6.8844</td> </tr> <tr> <td>Layer 1: Port FastEthernet0/4</td> <td>Layer 1: Port(s): FastEthernet0/2</td> </tr> </tbody> </table> <p>1. FastEthernet0/4 receives the frame.</p>	In Layers	Out Layers	Layer7	Layer7	Layer6	Layer6	Layer5	Layer5	Layer4	Layer4	Layer3	Layer3	Layer 2: Ethernet II Header 0050.0FC4.9341 >> 0002.17D6.8844	Layer 2: Ethernet II Header 0050.0FC4.9341 >> 0002.17D6.8844	Layer 1: Port FastEthernet0/4	Layer 1: Port(s): FastEthernet0/2
In Layers	Out Layers																
Layer7	Layer7																
Layer6	Layer6																
Layer5	Layer5																
Layer4	Layer4																
Layer3	Layer3																
Layer 2: Ethernet II Header 0050.0FC4.9341 >> 0002.17D6.8844	Layer 2: Ethernet II Header 0050.0FC4.9341 >> 0002.17D6.8844																
Layer 1: Port FastEthernet0/4	Layer 1: Port(s): FastEthernet0/2																
<p>El switch identifica la dirección MAC de origen como válida dentro de su tabla MAC, indicando que la trama es unicast. Luego, busca la dirección MAC de destino (0002.17D6.8844) en su tabla MAC para determinar el puerto de salida correspondiente, lo que permite decidir hacia dónde reenviar la trama.</p>	<p>At Device: Switch4 Source: RADIUS Destination: 67.20.20.190</p> <table border="1"> <thead> <tr> <th>In Layers</th> <th>Out Layers</th> </tr> </thead> <tbody> <tr><td>Layer7</td><td>Layer7</td></tr> <tr><td>Layer6</td><td>Layer6</td></tr> <tr><td>Layer5</td><td>Layer5</td></tr> <tr><td>Layer4</td><td>Layer4</td></tr> <tr><td>Layer3</td><td>Layer3</td></tr> <tr> <td>Layer 2: Ethernet II Header 0050.0FC4.9341 >> 0002.17D6.8844</td> <td>Layer 2: Ethernet II Header 0050.0FC4.9341 >> 0002.17D6.8844</td> </tr> <tr> <td>Layer 1: Port FastEthernet0/4</td> <td>Layer 1: Port(s): FastEthernet0/2</td> </tr> </tbody> </table> <p>1. The frame source MAC address was found in the MAC table of Switch. 2. This is a unicast frame. Switch looks in its MAC table for the destination MAC address.</p>	In Layers	Out Layers	Layer7	Layer7	Layer6	Layer6	Layer5	Layer5	Layer4	Layer4	Layer3	Layer3	Layer 2: Ethernet II Header 0050.0FC4.9341 >> 0002.17D6.8844	Layer 2: Ethernet II Header 0050.0FC4.9341 >> 0002.17D6.8844	Layer 1: Port FastEthernet0/4	Layer 1: Port(s): FastEthernet0/2
In Layers	Out Layers																
Layer7	Layer7																
Layer6	Layer6																
Layer5	Layer5																
Layer4	Layer4																
Layer3	Layer3																
Layer 2: Ethernet II Header 0050.0FC4.9341 >> 0002.17D6.8844	Layer 2: Ethernet II Header 0050.0FC4.9341 >> 0002.17D6.8844																
Layer 1: Port FastEthernet0/4	Layer 1: Port(s): FastEthernet0/2																
<p>El puerto de salida identificado es un puerto trunk, y el número de VLAN del puerto de entrada está permitido en ese trunk. Esto asegura que la trama pueda ser transmitida correctamente entre switches o hacia otros dispositivos a través del enlace trunk.</p>	<p>At Device: Switch4 Source: RADIUS Destination: 67.20.20.190</p> <table border="1"> <thead> <tr> <th>In Layers</th> <th>Out Layers</th> </tr> </thead> <tbody> <tr><td>Layer7</td><td>Layer7</td></tr> <tr><td>Layer6</td><td>Layer6</td></tr> <tr><td>Layer5</td><td>Layer5</td></tr> <tr><td>Layer4</td><td>Layer4</td></tr> <tr><td>Layer3</td><td>Layer3</td></tr> <tr> <td>Layer 2: Ethernet II Header 0050.0FC4.9341 >> 0002.17D6.8844</td> <td>Layer 2: Ethernet II Header 0050.0FC4.9341 >> 0002.17D6.8844</td> </tr> <tr> <td>Layer 1: Port FastEthernet0/4</td> <td>Layer 1: Port(s): FastEthernet0/2</td> </tr> </tbody> </table> <p>1. The outgoing port is a trunk port and the incoming port VLAN number is allowed in the trunk. Switch sends out the frame to that port.</p>	In Layers	Out Layers	Layer7	Layer7	Layer6	Layer6	Layer5	Layer5	Layer4	Layer4	Layer3	Layer3	Layer 2: Ethernet II Header 0050.0FC4.9341 >> 0002.17D6.8844	Layer 2: Ethernet II Header 0050.0FC4.9341 >> 0002.17D6.8844	Layer 1: Port FastEthernet0/4	Layer 1: Port(s): FastEthernet0/2
In Layers	Out Layers																
Layer7	Layer7																
Layer6	Layer6																
Layer5	Layer5																
Layer4	Layer4																
Layer3	Layer3																
Layer 2: Ethernet II Header 0050.0FC4.9341 >> 0002.17D6.8844	Layer 2: Ethernet II Header 0050.0FC4.9341 >> 0002.17D6.8844																
Layer 1: Port FastEthernet0/4	Layer 1: Port(s): FastEthernet0/2																
<p>Finalmente, el Switch4 reenvía la trama por el puerto FastEthernet0/2, dirigiéndola hacia el siguiente salto en la red con la intención de que llegue al dispositivo con la dirección MAC 0002.17D6.8844, completando así su función de reenvío en esta parte del proceso.</p>	<p>At Device: Switch4 Source: RADIUS Destination: 67.20.20.190</p> <table border="1"> <thead> <tr> <th>In Layers</th> <th>Out Layers</th> </tr> </thead> <tbody> <tr><td>Layer7</td><td>Layer7</td></tr> <tr><td>Layer6</td><td>Layer6</td></tr> <tr><td>Layer5</td><td>Layer5</td></tr> <tr><td>Layer4</td><td>Layer4</td></tr> <tr><td>Layer3</td><td>Layer3</td></tr> <tr> <td>Layer 2: Ethernet II Header 0050.0FC4.9341 >> 0002.17D6.8844</td> <td>Layer 2: Ethernet II Header 0050.0FC4.9341 >> 0002.17D6.8844</td> </tr> <tr> <td>Layer 1: Port FastEthernet0/4</td> <td>Layer 1: Port(s): FastEthernet0/2</td> </tr> </tbody> </table> <p>1. FastEthernet0/2 sends out the frame.</p>	In Layers	Out Layers	Layer7	Layer7	Layer6	Layer6	Layer5	Layer5	Layer4	Layer4	Layer3	Layer3	Layer 2: Ethernet II Header 0050.0FC4.9341 >> 0002.17D6.8844	Layer 2: Ethernet II Header 0050.0FC4.9341 >> 0002.17D6.8844	Layer 1: Port FastEthernet0/4	Layer 1: Port(s): FastEthernet0/2
In Layers	Out Layers																
Layer7	Layer7																
Layer6	Layer6																
Layer5	Layer5																
Layer4	Layer4																
Layer3	Layer3																
Layer 2: Ethernet II Header 0050.0FC4.9341 >> 0002.17D6.8844	Layer 2: Ethernet II Header 0050.0FC4.9341 >> 0002.17D6.8844																
Layer 1: Port FastEthernet0/4	Layer 1: Port(s): FastEthernet0/2																

El dispositivo RADIUS recibe una trama Ethernet por el puerto FastEthernet0. Esta trama contiene una cabecera Ethernet de Capa 2, con la dirección MAC de origen 0050.0FC4.9341 (del dispositivo DHCP) y como destino la MAC 0002.17D6.8844 (perteneciente a RADIUS). En la Capa 3 (IP), el paquete tiene como dirección IP de origen 67.20.20.190 y como destino 67.20.20.50, con un tipo de mensaje ICMP 0, lo que indica que se trata de una respuesta de eco (Echo Reply).

At Device: RADIUS
Source: RADIUS
Destination: 67.20.20.190

In Layers

Layer7
Layer6
Layer5
Layer4
Layer 3: IP Header Src. IP: 67.20.20.190,
Dest. IP: 67.20.20.50 ICMP Message Type:
0
Layer 2: Ethernet II Header
0050.0FC4.9341 >> 0002.17D6.8844
Layer 1: Port FastEthernet0

Out Layers

Layer7
Layer6
Layer5
Layer4
Layer3
Layer2
Layer1

1. FastEthernet0 receives the frame.

El dispositivo RADIUS verifica que la dirección MAC de destino de la trama coincida con la suya propia, una dirección de broadcast o multicast, lo cual permite aceptar la trama. Una vez validado esto, procede a desencapsular la unidad de datos de protocolo (PDU) del marco Ethernet, extrayendo la información contenida para ser procesada en las capas superiores. Esto corresponde a la transición de la Capa 2 a la Capa 3.

At Device: RADIUS
Source: RADIUS
Destination: 67.20.20.190

In Layers

Layer7
Layer6
Layer5
Layer4
Layer 3: IP Header Src. IP: 67.20.20.190,
Dest. IP: 67.20.20.50 ICMP Message Type:
0
Layer 2: Ethernet II Header
0050.0FC4.9341 >> 0002.17D6.8844
Layer 1: Port FastEthernet0

Out Layers

Layer7
Layer6
Layer5
Layer4
Layer3
Layer2
Layer1

1. The packet's destination IP address matches the device's IP address or the broadcast address. The device de-encapsulates the packet.
2. The packet is an ICMP packet. The ICMP process processes it.
3. The ICMP process received an Echo Reply message.
4. The Ping process received an Echo Reply message.

En la capa de red (Capa 3), el dispositivo RADIUS identifica que la dirección IP de destino del paquete (67.20.20.190) coincide con su propia dirección IP. El paquete se valida como un mensaje ICMP de tipo 0 (Echo Reply), el cual corresponde a una respuesta de ping. El proceso ICMP dentro del dispositivo procesa esta respuesta, confirmando que se ha recibido satisfactoriamente un mensaje de tipo "Echo Reply". Finalmente, el proceso del comando ping en RADIUS reconoce que la respuesta ha sido recibida desde el host 67.20.20.50, completando así la prueba de conectividad.

At Device: RADIUS
Source: RADIUS
Destination: 67.20.20.190

In Layers

Layer7
Layer6
Layer5
Layer4
Layer 3: IP Header Src. IP: 67.20.20.190,
Dest. IP: 67.20.20.50 ICMP Message Type:
0
Layer 2: Ethernet II Header
0050.0FC4.9341 >> 0002.17D6.8844
Layer 1: Port FastEthernet0

Out Layers

Layer7
Layer6
Layer5
Layer4
Layer3
Layer2
Layer1

1. The frame's destination MAC address matches the receiving port's MAC address, the broadcast address, or a multicast address.
2. The device decapsulates the PDU from the Ethernet frame.

En el presente escenario de red, se analiza el intercambio de tráfico ICMP entre dos dispositivos: el servidor RADIUS, con dirección IP 67.20.20.190 y MAC 0050.0FC4.9341, y el servidor DHCP, con IP 67.20.20.50 y MAC 0002.17D6.8844. El tipo de tráfico involucrado corresponde a mensajes ICMP, específicamente solicitudes y respuestas de eco (Echo Request y Echo Reply), comúnmente utilizados para pruebas de conectividad.

El proceso inicia cuando el servidor RADIUS genera un paquete ICMP tipo 8 (Echo Request) dirigido al servidor DHCP. Este paquete se encapsula en una trama Ethernet II, con la dirección MAC de origen correspondiente al RADIUS y la MAC de destino al DHCP. El paquete ingresa al Switch4 por el puerto FastEthernet0/2 y, gracias a la tabla de direcciones MAC del switch, se reenvía por el puerto FastEthernet0/4, que está configurado como trunk y permite el paso del VLAN correspondiente.

Una vez que el paquete llega al servidor DHCP por el puerto FastEthernet0/0, se verifica que la dirección MAC de destino coincida con la del dispositivo. El paquete se desencapsula y se procesa en la capa de red, confirmando que la dirección IP de destino es la propia. Al tratarse de un mensaje ICMP tipo 8, el servidor DHCP genera una respuesta tipo 0 (Echo Reply), indicando que ha recibido correctamente la solicitud de eco.

La respuesta ICMP se encapsula nuevamente en una trama Ethernet II, esta vez con la MAC de origen del DHCP y la MAC de destino del RADIUS. El paquete se envía de regreso al Switch4 por el puerto FastEthernet0/0. El switch, al recibir el paquete por FastEthernet0/4, consulta su tabla MAC y determina que debe reenviarlo por FastEthernet0/2 hacia el servidor RADIUS, validando que el puerto de salida permite el VLAN del paquete.

En la Red Real

El análisis y monitoreo del tráfico en una red real es fundamental para comprender el comportamiento de los protocolos y la interacción entre los dispositivos conectados. En esta etapa del informe, se abordará la captura y análisis de paquetes utilizando herramientas especializadas que permiten observar en detalle el flujo de datos y los procesos subyacentes en la red.

Usando Wireshark

En esta fase del laboratorio se utilizará Wireshark, un analizador de protocolos de red multiplataforma, para capturar y examinar el tráfico de red en tiempo real. El objetivo principal es profundizar en el análisis de las diferentes capas del modelo OSI mediante la observación detallada de los paquetes transmitidos durante consultas HTTP. Se aplicarán filtros específicos para identificar y analizar protocolos como TCP, IP, ICMP y HTTP, evaluando el proceso de encapsulamiento y desencapsulamiento de datos en cada capa.

Se elaboró un video explicativo en el que se detalla el funcionamiento de la interfaz gráfica de Wireshark. Este video tiene como finalidad facilitar la comprensión de las herramientas que ofrece el programa, mostrando paso a paso cómo navegar por sus distintas secciones, identificar los paquetes capturados y cómo interpretar la información que se presenta en cada uno de ellos. También se enseñó cómo crear filtros personalizados que permiten aislar ciertos protocolos o direcciones IP, lo cual es esencial para un análisis más preciso y enfocado.

[Mira el Video: “Explorando Wireshark: Interfaz, Filtros y Primeros Pasos”]

(<https://drive.google.com/file/d/1Etai3NfV4Hu4kK7rqFdbOuhWJCSDPfvx/view?usp=sharing>)

(3.3. *The Main Window*, s. f.; 6.3. *Filtering Packets While Viewing*, s. f.; 6.6. *Defining And Saving Filters*, s. f.)

Adicionalmente, se realizó otro video de prueba en el que se accedió a dos páginas web específicas: scielo.org y scielo.org.co. Durante esta prueba, se utilizó Wireshark para capturar el tráfico generado por estas visitas, permitiendo observar cómo se producen las solicitudes y respuestas HTTP, así como las negociaciones TCP previas. Este ejercicio tuvo como objetivo evidenciar en tiempo real el funcionamiento de los protocolos involucrados en una conexión web, y cómo se pueden aplicar los filtros configurados anteriormente para aislar los paquetes correspondientes a cada sitio. Estas pruebas prácticas complementan el análisis teórico y fortalecen la comprensión del comportamiento del tráfico en redes reales.

[Mira el Video: “Investigación de Tráfico HTTP Análisis de Paquetes en Wireshark”]
(https://drive.google.com/file/d/1ymbgir_77MVjtkLVwTp7igfHdXAVWH35/view?usp=sharing)

Tarjetas de Red

En esta fase se realizará un análisis comparativo de las tarjetas de red físicas y virtuales, recopilando parámetros críticos como direcciones MAC, direcciones IP (IPv4 e IPv6), velocidades de enlace, fabricantes y estadísticas de tráfico. Este análisis permitirá evaluar las diferencias operativas y de rendimiento entre dispositivos reales y virtualizados, aportando una visión integral del entorno de red.

Además, el estudio se llevó a cabo utilizando dos máquinas físicas del laboratorio y dos máquinas virtuales configuradas con los sistemas operativos Slackware y Solaris. Esta combinación permitió contrastar directamente el comportamiento de las interfaces de red en

entornos físicos y virtuales, enriqueciendo el análisis con datos provenientes de plataformas heterogéneas y ampliando la validez de las conclusiones obtenidas.

Figura 11*Configuración de red de máquina física con adaptador Realtek*

IP assignment:	Automatic (DHCP)	Edit
DNS server assignment:	Automatic (DHCP)	Edit
Aggregated link speed (Receive/Transmit):	100/100 (Mbps)	Copy
Link-local IPv6 address:	fe80::5248:7897:ac2:1751%4	
IPv4 address:	10.2.67.103	
IPv4 default gateway:	10.2.65.1	
IPv4 DNS servers:	10.2.65.2 (Unencrypted) 10.2.65.61 (Unencrypted) 10.2.65.62 (Unencrypted)	
Primary DNS suffix:	is.escuelaing.edu.co	
Manufacturer:	Realtek	
Description:	Realtek PCIe GbE Family Controller #2	
Driver version:	10.54.1111.2021	
Physical address (MAC):	88:AE:DD:5E:29:15	

Nota. La imagen muestra los parámetros de red de una máquina física equipada con un adaptador Realtek PCIe GbE Family Controller. Se observa asignación automática de IP y DNS mediante DHCP, velocidad de enlace de 100 Mbps, dirección IPv4 10.2.67.103, dirección MAC 80:4E:DD:5E:29:15, y detalles del controlador de red.

Figura 12*Configuración de red de máquina física con adaptador Intel*

IP assignment:	Automatic (DHCP)	Edit
DNS server assignment:	Automatic (DHCP)	Edit
Aggregated link speed (Receive/Transmit):	100/100 (Mbps)	Copy
Link-local IPv6 address:	fe80::951d:b261:3184:e1d9%17	
IPv4 address:	10.2.67.106	
IPv4 default gateway:	10.2.65.1	
IPv4 DNS servers:	10.2.65.2 (Unencrypted) 10.2.65.61 (Unencrypted) 10.2.65.62 (Unencrypted)	
Primary DNS suffix:	is.escuelaing.edu.co	
Manufacturer:	Intel	
Description:	Intel(R) Ethernet Connection (7) I219-LM	
Driver version:	12.19.2.62	
Physical address (MAC):	30:13:8B:6A:19:17	

Nota. La imagen presenta la configuración de red de una máquina física con adaptador Intel(R) Ethernet Connection (7) I219-LM. Incluye asignación automática de IP y DNS, velocidad de enlace de 100 Mbps, dirección IPv4 10.2.65.121, dirección MAC 30:1B:38:64:B1:97, y versión del controlador instalada.

Figura 13*Identificación del adaptador de red Intel en máquina virtual*

```

root@darkstar:~# lspci | grep -i ethernet
02:01.0 Ethernet controller: Intel Corporation 82545EM Gigabit Ethernet Controller (Copper) (rev 01)
root@darkstar:~# _

```

Nota. La imagen muestra el resultado del comando `lspci` en una máquina virtual con sistema operativo Slackware. Se identifica un adaptador de red Intel Corporation 82545EM Gigabit Ethernet Controller (Copper), modelo utilizado comúnmente en entornos virtualizados. Esta información permite conocer el fabricante y modelo del controlador físico o virtual asignado.

Figura 14*Parámetros de red y tráfico en interfaz Ethernet de máquina virtual*

```

root@darkstar:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.2.77.182 netmask 255.255.0.0 broadcast 10.2.255.255
    inet6 fe80::20c:29ff:fe9d:bead prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:9d:be:ad txqueuelen 1000 (Ethernet)
    RX packets 6444 bytes 1005225 (981.6 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 5 bytes 446 (446.0 B)
    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 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@darkstar:~#

```

Nota. La imagen presenta la configuración de red obtenida mediante el comando `ifconfig` en Slackware. Se observa la interfaz `eth0` con dirección IPv4 10.2.77.182, dirección MAC 00:0c:29:9d:be:ad, dirección IPv6 fe80::20c:29ff:fe9d:bead, y una MTU de 1500. Se incluyen estadísticas de tráfico con 981.6 KiB recibidos y 446 bytes transmitidos, lo cual permite evaluar el uso de red de la máquina virtual.

Figura 15*Velocidad y estado físico de la interfaz de red en Solaris*

```

root@solaris:~# dladm show-phys
LINK          MEDIA      STATE      SPEED  DUPLEX    DEVICE
net0          Ethernet  up         1000   full     e1000g0
root@solaris:~# █

```

Nota. La imagen muestra el resultado del comando `dladm show-phys` ejecutado en una máquina virtual con Solaris. Se detalla que la interfaz `net0` es de tipo Ethernet, se encuentra en estado activo (`up`), opera a una velocidad de 1000 Mbps (1 Gbps), en modo dúplex completo, y utiliza el dispositivo `e1000g0`. Esta información es útil para verificar la capacidad y estado del enlace de red.

Figura 16*Configuración IP y dirección MAC en máquina virtual con Solaris*

```

root@solaris:~# ifconfig -a
lo0: flags=2001000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv4,VIRTUAL> mtu 8232 index
1
    inet 127.0.0.1 netmask ffffffff
net0: flags=100001000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4,PHYSRUNNING> mtu 15
00 index 2
    inet 10.2.77.180 netmask ffff0000 broadcast 10.2.255.255
    ether 0:c:29:4f:ee:1b
lo0: flags=2002000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv6,VIRTUAL> mtu 8252 index
1
    inet6 ::1/128
net0: flags=120002004841<UP,RUNNING,MULTICAST,DHCP,IPv6,PHYSRUNNING> mtu 1500 in
dex 2
    inet6 fe80::20c:29ff:fe4f:ee1b/10
    ether 0:c:29:4f:ee:1b
root@solaris:~# █

```

Nota. Captura del comando `ifconfig -a` en una máquina virtual Solaris, donde se observa la interfaz `net0` con dirección IPv4 10.2.77.180, dirección MAC 0:c:29:4f:ee:1b y dirección IPv6 fe80::20c:29ff:fe4f:ee1b/10. Se incluyen los parámetros del loopback y se confirma que la interfaz física está activa con DHCP habilitado. Esta información permite identificar las direcciones asignadas y el estado operativo de la red.

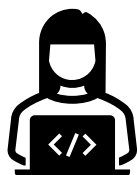
Software Base

Programación en Shell – Unix

En esta actividad se desarrollaron una serie de scripts en Shell orientados a la administración básica del sistema operativo en entornos Unix, específicamente en distribuciones como Slackware, Solaris y CentOS. El objetivo principal fue fortalecer el conocimiento práctico sobre el funcionamiento interno del sistema, automatizando tareas comunes de administración mediante programación en Shell.

Figura 17

Listar y clasificar archivos según criterios como fecha, tamaño y tipo.

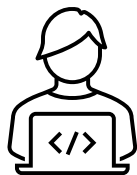


filtrar_listar.sh

Nota. Proporciona un menú interactivo para sistemas Unix como Slackware o Solaris, permitiendo al usuario realizar tareas comunes sobre archivos de texto, como buscar archivos por nombre en una ruta, contar cuántas veces aparece una palabra en un archivo, mostrar líneas específicas (primeras o últimas), contar líneas totales, y buscar palabras en múltiples archivos mostrando sus ubicaciones y frecuencia.

Figura 18

Buscar archivos y palabras dentro de archivos, mostrando coincidencias y ubicaciones.

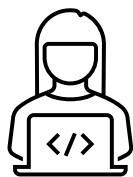


buscar_archivo.sh

Nota. Ofrece un menú interactivo que permite realizar tareas comunes de búsqueda y análisis de archivos en sistemas Unix como Slackware o Solaris.

Figura 19

Revisar archivos de log del sistema y filtrar eventos por palabra clave.

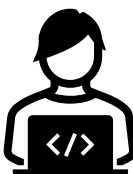


archivo_log.sh

Nota. Muestra las últimas 15 líneas de tres archivos de log del sistema y permite filtrar esas líneas por una palabra específica.

Figura 20

Automatizar la creación de grupos en sistemas Unix.



n_grupo.sh

Nota. permite crear grupos de usuarios en sistemas como Slackware o Solaris, verificando si el grupo ya existe y adaptándose al sistema operativo para usar el comando adecuado.

Figura 21

Automatizar la creación de usuarios, grupos y permisos en sistemas Unix.



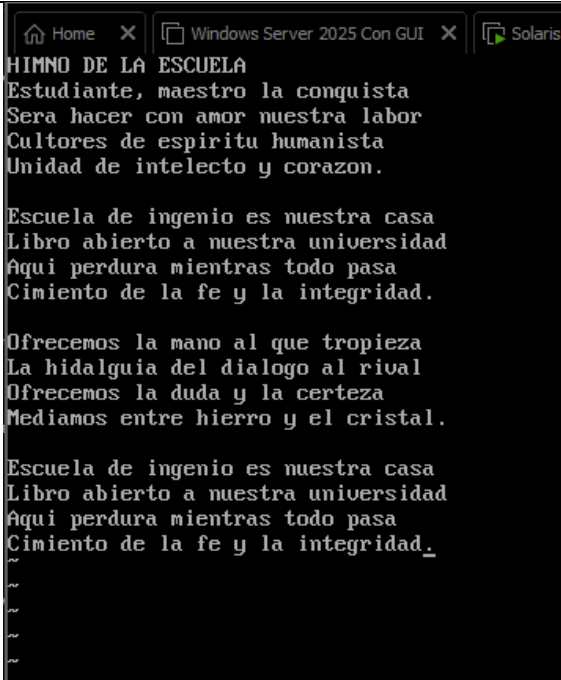
n_usuario.sh

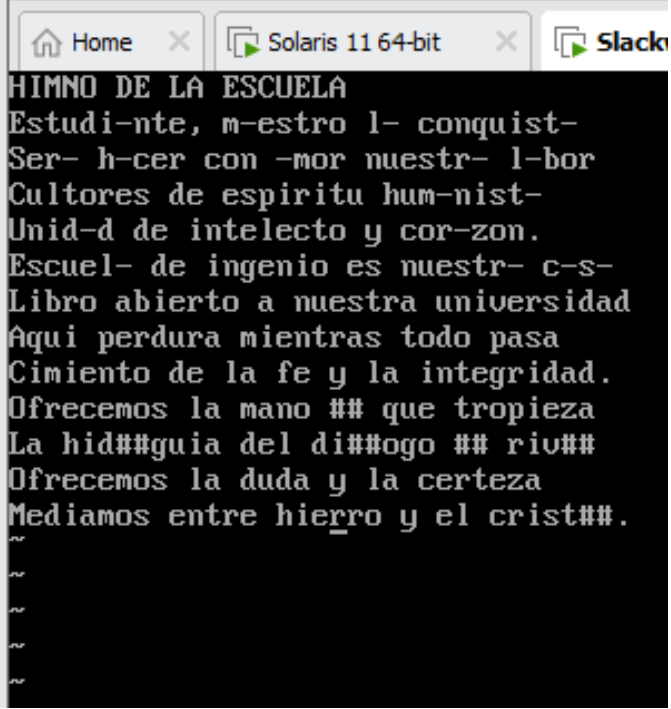
Nota. Crea un nuevo usuario con su grupo, directorio personal, shell, y permisos personalizados.

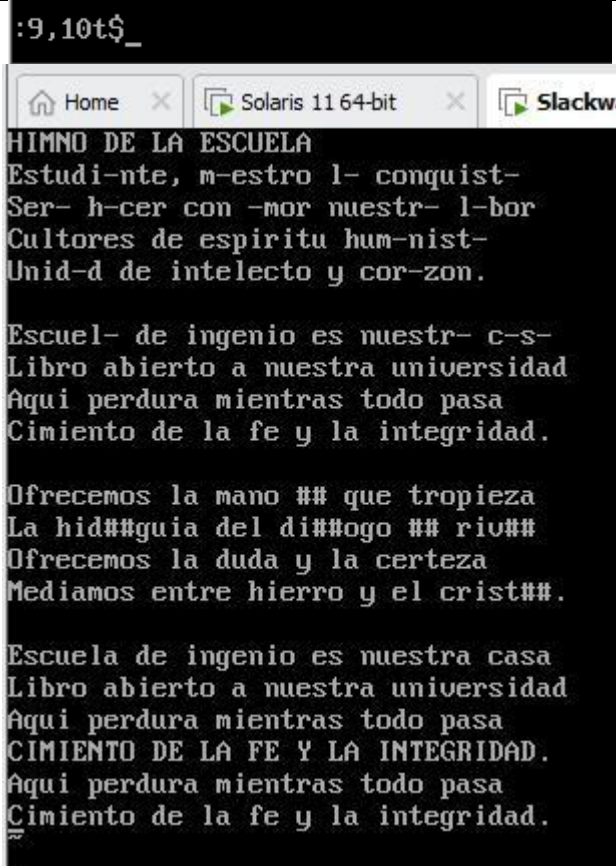
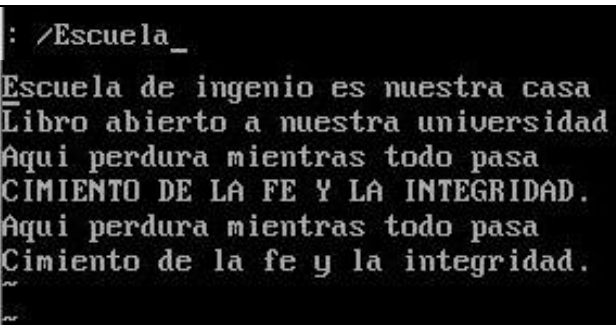
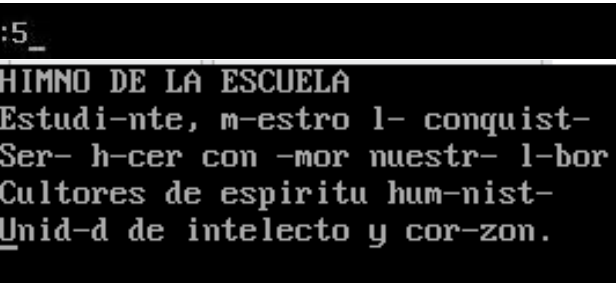
Editor VI en Linux/Unix

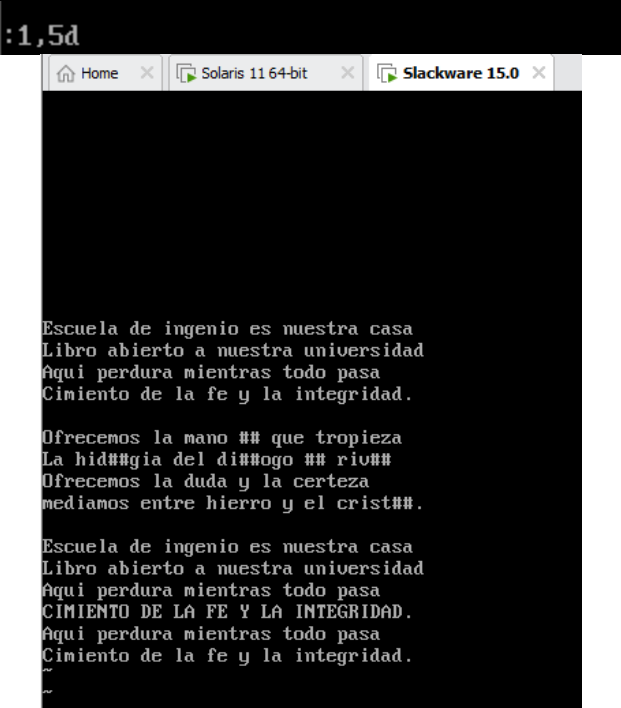
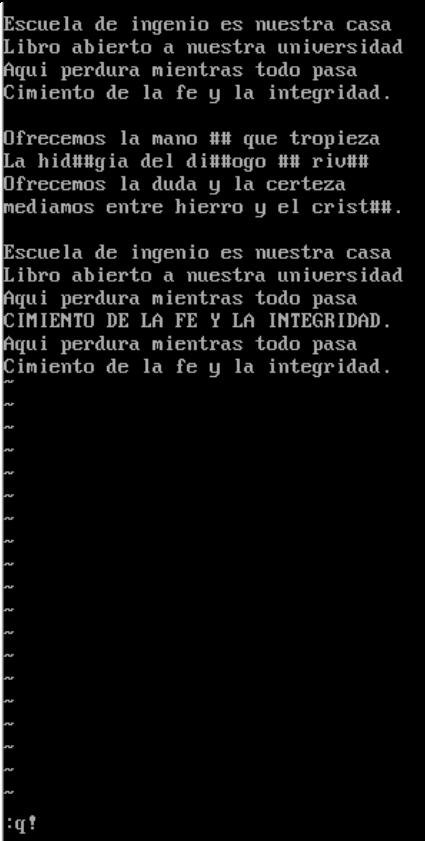
A continuación, se presentarán un conjunto de operaciones que incluyen: la creación y edición de un archivo, la inserción de texto línea por línea, la búsqueda y reemplazo de caracteres o palabras, la eliminación y copia de líneas, la conversión de texto a mayúsculas, así como la navegación dentro del documento. También se mostrará cómo guardar cambios, salir del editor y trabajar con diferentes escenarios de edición.

De esta manera, el lector podrá comprender y ejercitar de manera práctica los comandos más importantes del editor VI, desarrollando habilidades útiles para el manejo de archivos en entornos Linux/Unix.

Actividad/Acción/Tarea	Detalles relevantes
Usa el editor VI para crear un archivo. Documenta los comandos utilizados.	<code>root@darkstar:~# vi himno.txt_</code>
Ingresa el siguiente texto y documenta los comandos utilizados.	

	<pre> HIMNO DE LA ESCUELA^M Estudi-nte, m-estro l- conquist- Ser- h-cer con -mor nustr- l-bor Cultores de espiritu hum-nist- Unid-d de intelecto y cor-zon. Escuel- de ingenio es nustr- c-s- Libro abierto a nuestra universidad Aqui perdura mientras todo pasa Cimiento de la fe y la integridad. Ofrecemos la mano ## que tropieza La hid##guia del di##ogo ## rivo## Ofrecemos la duda y la certeza Mediamos entre hierro y el crist##. Escuela de ingenio es nuestra casa Libro abierto a nuestra universidad Aqui perdura mientras todo pasa Cimiento de la fe y la integridad. ~ ~ </pre>
<p>Elimina las últimas cuatro líneas del documento usando un solo comando.</p>	<pre> :\$-3,\$d_ </pre>  <p>The screenshot shows a terminal window with tabs for 'Home', 'Solaris 11 64-bit', and 'Slack'. The command '\$-3,\$d_' has been entered, and the output is the same text as the previous block, but with the last four lines removed.</p>
<p>Deshaz el comando anterior.</p>	<pre> :u </pre>

<p>Copia las últimas dos líneas del segundo párrafo al final del archivo.</p>	 <pre> :9,10t\$_ HIMNO DE LA ESCUELA Estudi-nte, m-estro l- conquist- Ser- h-cer con -mor nustr- l-bor Cultores de espiritu hum-nist- Unid-d de intelecto y cor-zon. Escuel- de ingenio es nustr- c-s- Libro abierto a nuestra universidad Aqui perdura mientras todo pasa Cimiento de la fe y la integridad. Ofrecemos la mano ## que tropieza La hid##guia del di##ogo ## riu## Ofrecemos la duda y la certeza Mediamos entre hierro y el crist##. Escuela de ingenio es nuestra casa Libro abierto a nuestra universidad Aqui perdura mientras todo pasa CIMIENTO DE LA FE Y LA INTEGRIDAD. Aqui perdura mientras todo pasa Cimiento de la fe y la integridad. </pre>						
<p>Busca la palabra “Escuela” dentro del texto.</p>	 <pre> : /Escuela_ Escuela de ingenio es nuestra casa Libro abierto a nuestra universidad Aqui perdura mientras todo pasa CIMIENTO DE LA FE Y LA INTEGRIDAD. Aqui perdura mientras todo pasa Cimiento de la fe y la integridad. </pre>						
<p>Muévete a la línea 5 del texto usando un comando.</p>	 <pre> :5_ HIMNO DE LA ESCUELA Estudi-nte, m-estro l- conquist- Ser- h-cer con -mor nustr- l-bor Cultores de espiritu hum-nist- Unid-d de intelecto y cor-zon. </pre>						
<p>Crea una tabla resumen con comandos de VI.</p>	<table border="1"> <thead> <tr> <th data-bbox="618 1766 992 1808">Acción</th><th data-bbox="992 1766 1365 1808">Comando</th></tr> </thead> <tbody> <tr> <td data-bbox="618 1808 992 1850">Entrar en modo inserción</td><td data-bbox="992 1808 1365 1850">I</td></tr> <tr> <td data-bbox="618 1850 992 1877">Salir al modo comando</td><td data-bbox="992 1850 1365 1877">Esc</td></tr> </tbody> </table>	Acción	Comando	Entrar en modo inserción	I	Salir al modo comando	Esc
Acción	Comando						
Entrar en modo inserción	I						
Salir al modo comando	Esc						

<p>Vuelve a abrir el archivo y elimina las primeras cinco líneas.</p>	
<p>Sal del archivo sin guardar.</p>	

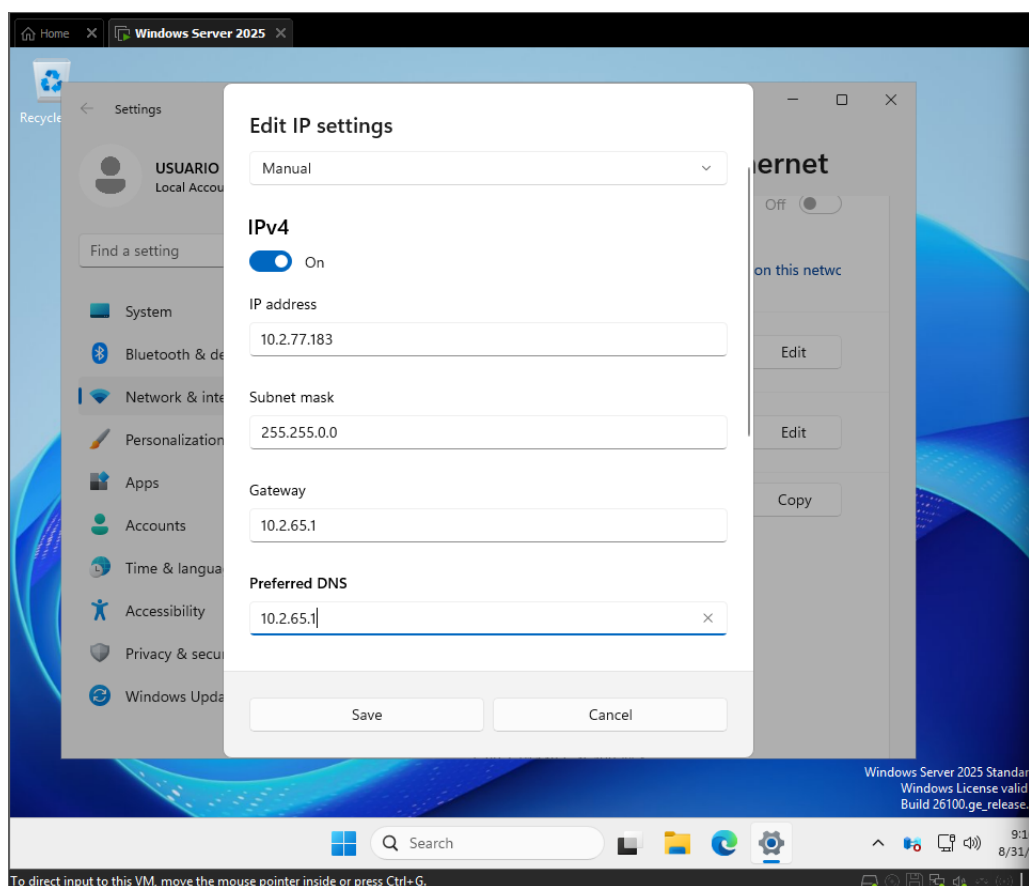
Despliegue de Máquinas Virtuales

Para el desarrollo de esta práctica se llevará a cabo el despliegue de varias máquinas virtuales con diferentes sistemas operativos. A continuación, se detallan los procedimientos necesarios para su creación, configuración de red, verificación de comunicación entre ellas y comprobación de acceso a internet.

El proceso de instalación de cada una de las máquinas virtuales ya fue explicado en el Laboratorio 1. Siguiendo esos mismos pasos de instalación, en esta práctica se procederá a configurar la red de cada máquina virtual y a comprobar su conectividad, realizando pruebas de ping entre ellas y verificando también el acceso a internet.

Figura 17

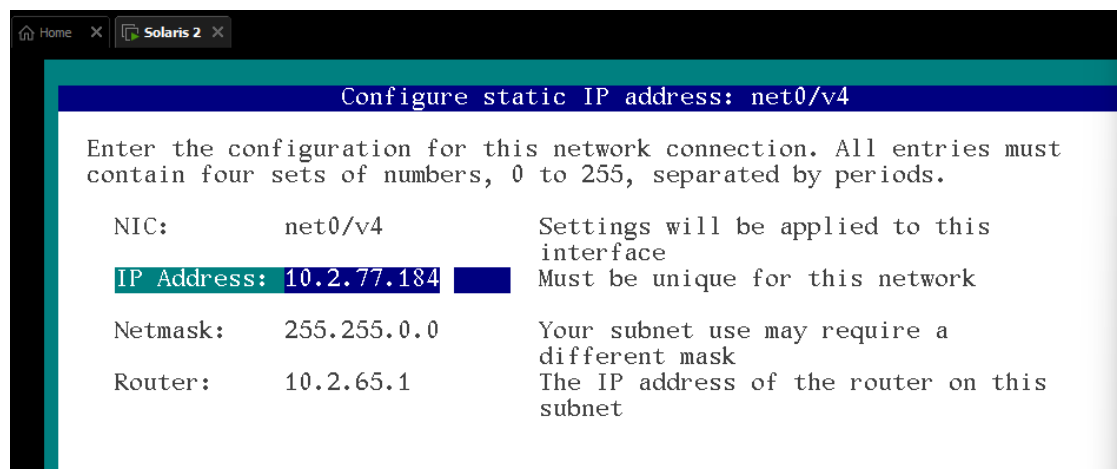
Configuración de red de la segunda máquina virtual de Windows.



Nota. Captura de la configuración manual de red en una máquina virtual con Windows Server 2025, donde se asigna la dirección IPv4 10.2.77.183 con máscara de subred 255.255.0.0. Se establece como puerta de enlace y servidor DNS preferido la dirección 10.2.65.1, lo que permite la comunicación dentro de la red y el acceso a internet. Esta configuración asegura que la máquina virtual pueda interactuar con otras dentro del mismo segmento de red y validar conectividad mediante pruebas de ping.

Figura 18

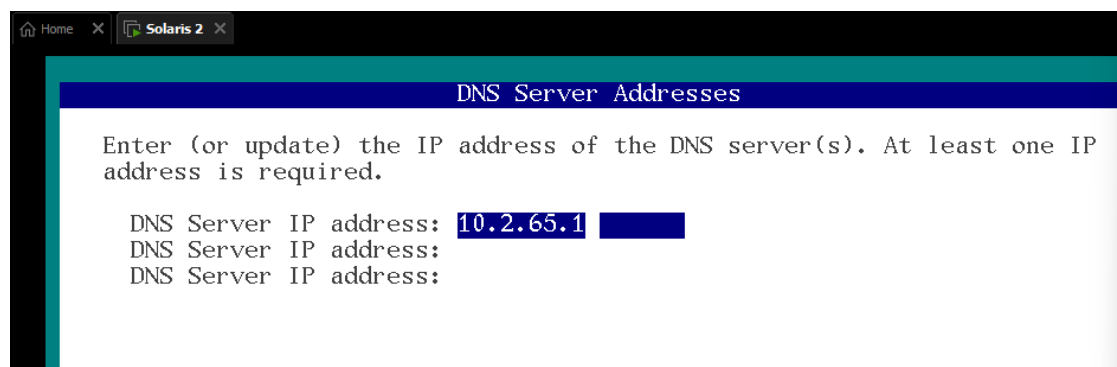
Configuración de red de la segunda máquina virtual de Solaris



Nota. Captura del proceso de configuración de dirección IP estática en una máquina virtual con Solaris, donde la interfaz net0/v4 recibe la dirección IPv4 10.2.77.184, con máscara de subred 255.255.0.0. Esta configuración permite integrar la máquina dentro de la red definida, facilitando la comunicación con otras máquinas virtuales y el acceso a internet a través del router especificado.

Figura 19

Configuración de configuración de servidores DNS



Nota. Captura de la configuración de servidores DNS en una máquina virtual con Solaris, donde se define la dirección 10.2.65.1 como servidor DNS principal.

Figura 20

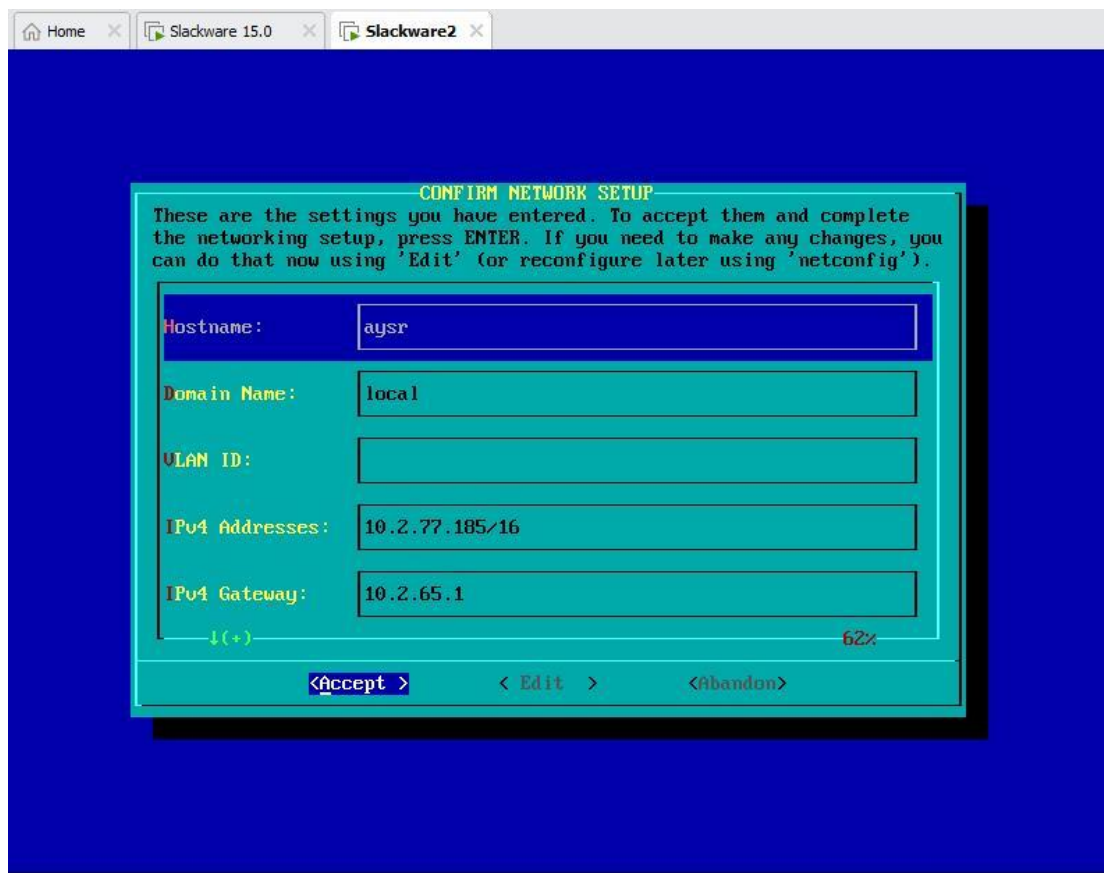
Prueba de red de la segunda máquina virtual de Solaris a la otra maquina Solaris.

```
root@solaris:~# ping 10.2.77.180
10.2.77.180 is alive
root@solaris:~# █
```

Nota. Captura de la prueba de conectividad en una máquina virtual con Solaris, donde mediante el comando ping se verifica la comunicación con otra máquina de la misma red. La respuesta positiva confirma que la configuración de la interfaz de red, la dirección IP y el gateway fueron asignados correctamente, permitiendo la interacción entre ambas máquinas virtuales.

Figura 21

Configuración de red de la segunda máquina virtual de Slackware



Nota. Configuración de red en Slackware, donde se asigna la dirección IP estática 10.2.77.185 con su respectiva máscara de subred y puerta de enlace.

Figura 22

Prueba de red de la segunda máquina virtual de Slackware a la otra máquina Slackware.

```

root@aysr:~# ping 10.2.77.182
PING 10.2.77.182 (10.2.77.182) 56(84) bytes of data.
64 bytes from 10.2.77.182: icmp_seq=1 ttl=64 time=0.428 ms
64 bytes from 10.2.77.182: icmp_seq=2 ttl=64 time=0.647 ms
64 bytes from 10.2.77.182: icmp_seq=3 ttl=64 time=0.662 ms
^C
--- 10.2.77.182 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2041ms
rtt min/avg/max/mdev = 0.428/0.579/0.662/0.106 ms
root@aysr:~#

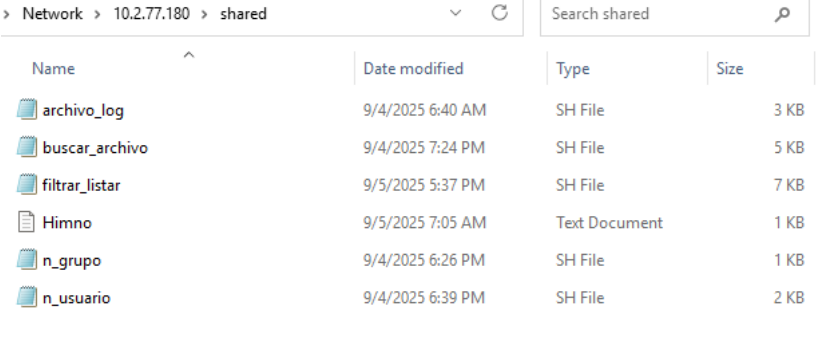
```

Nota. Prueba de red en Slackware, donde la segunda máquina virtual con dirección IP 10.2.77.185 realiza un ping exitoso hacia la primera máquina Slackware dentro de la misma red. El resultado confirma la correcta comunicación entre ambas máquinas con configuraciones de red estáticas.

Compartir Archivos

En esta parte de la práctica se trabajará con la compartición de archivos como un servicio fundamental en entornos empresariales. Para ello, se configurará un servidor de archivos en Solaris utilizando SMB/SAMBA, con el fin de permitir que diferentes sistemas operativos puedan acceder a un directorio común y compartir información.

Actividad/Acción/Tarea	Detalles relevantes
Se debe instalar Samba en la máquina de Solaris con el comando <code>pkg install samba</code> , para compartir carpetas a través del protocolo SMB.	<pre> root@solaris:~# pkg install samba Paquetes que instalar: 4 Mediadores que cambiar: 1 Servicios que cambiar: 2 Crear entorno de inicio: No Crear copia de seguridad de entorno de inicio: No DOWNLOAD PKGS FILES XFER (MB) SPEED Completado 4/4 996/996 19.5/19.5 2.2M/s PHASE ITEMS Instalando acciones nuevas 1214/1214 Actualizando base de datos de estado de paquete Listo Actualizando caché de paquete 0/0 Actualizando estado de imagen Listo Creando base de datos de búsqueda rápida en proceso \Loading smf(7) service d Creando base de datos de búsqueda rápida en proceso /Aug 29 05:59:26 solaris sendmail[1185]: My unqualified host name (solaris) unknown; sleeping for retry Creando base de datos de búsqueda rápida en proceso / Creando base de datos de búsqueda rápida Listo Actualizando caché de paquete 1/1 </pre>
Posteriormente, se debe crear la carpeta que se desea compartir y se asignan permisos de lectura, escritura y ejecución para todos los usuarios.	<pre> root@solaris:~# mkdir -p /path/to/shared/directory root@solaris:~# chmod -R 0777 /path/to/shared/directory </pre>

<p>El archivo <code>/etc/samba/smb.conf</code> se configura para indicar a Samba qué directorios del sistema se van a compartir en la red y en qué condiciones.</p>	<pre>workgroup = WORKGROUP server string = Solaris File Server security = user map to guest = Bad User dns proxy = no [shared] path = /path/to/shared/directory valid users = @users read only = no guest ok = yes</pre>
<p>Ahora se procede a crear un usuario en Samba llamado <code>aurora</code> y asignarle una contraseña para poder autenticarse al acceder a los recursos compartidos.</p>	<pre>root@solaris:~# smbpasswd -a aurora New SMB password: Retype new SMB password: Added user aurora. root@solaris:~# █</pre>
<p>Se debe habilitar al usuario de Samba previamente creado, con el fin de que este pueda conectarse al servidor de archivos.</p>	<pre>root@solaris:~# smbpasswd -e aurora Enabled user aurora. root@solaris:~# █</pre>
<p>Se debe habilitar e iniciar el servicio de Samba en Solaris con el comando <code>svcadm enable samba</code>. Para verificar el estado del servicio Samba se usa el comando <code>svcs -xv samba</code>, en este caso muestra online, lo que significa que el servidor SMB ya está corriendo correctamente.</p>	<pre>root@solaris:~# svcadm enable samba root@solaris:~# svcs -xv samba svc:/network/samba:default (SMB file server) Estado: online desde 29 de agosto de 2025, 6:28:46 -05 Consulte: man -M /usr/share/man -s 4 smb.conf Consulte: man -M /usr/share/man -s 1m smbmbd Consulte: /var/svc/log/network-samba:default.log Impacto: ninguno. root@solaris:~# █</pre>
<p>Para probar desde la máquina de Windows, en el explorador, en la barra de ruta se pone <code>\\10.2.77.180\shared</code>, la cual es la IP de Solaris, se debe ingresar el usuario y contraseña del usuario creado anteriormente en Samba</p>	

Para probar desde slackware, se debe instalar Samba, para esto es esencial que se modifique el archivo `/etc/slackpkg/slackpkg.conf`, aquí se debe modificar `CHECKMD5` y `CHECKGPG` ambos deben quedar en off

```
# When the files will be downloaded and the operation (install/upgrade)
# performed one by one. Default=on
DOWNLOAD_ALL=on

# Enables (on) or disables (off) the dialog interface in slackpkg. Default=on
DIALOG=on

# Enables (on) or disables (off) the non-interactive mode. If set to "on",
# slackpkg will run without asking the user anything, and answer all questions
# with DEFAULT_ANSWER. If you do any upgrades using this mode, you'll need to
# run "slackpkg new-config" later to find and merge any .new files.
BATCH=off

# Default answer to slackpkg questions. Can be "y" or "n".
DEFAULT_ANSWER=n

# Slackpkg allows a template to "include" the packages specified in another
# template. This option enables (on) or disables (off) the parsing of
# any "#include" directives in template files. Default=on
USE_INCLUDES=on

# Enables a spinning bar as visual feedback when slackpkg is making its
# internal lists and some other operations. Default=on
SPINNING=on

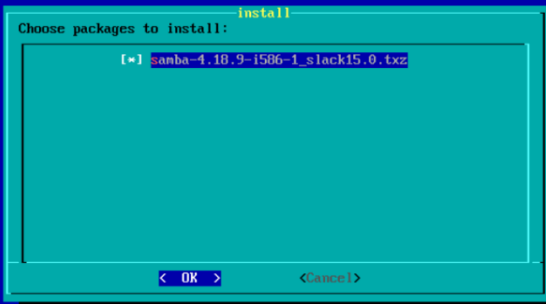
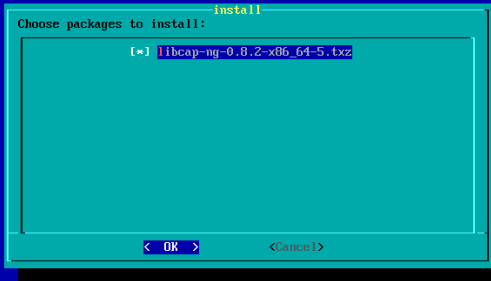
# Max number of characters that "dialog" command can handle.
# If unset, this variable will be 19500 (the number that works on
# Slackware 10.2)
DIALOG_MAXARGS=139000

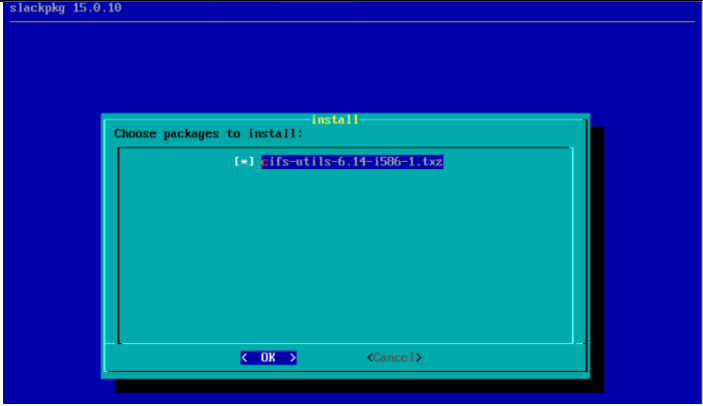
#
# The MIRROR is set from /etc/slackpkg/mirrors
# You only need to uncomment the selected mirror.
# Uncomment one mirror only.
#
/etc/slackpkg/slackpkg.conf: 157 lines, 6111 characters.
root@darkstar:~#
```

Se debe elegir un mirror, para esto se debe editar el archivo `/etc/slackpkg/mirrors`, dentro de archivo, se debe descomentar la línea que corresponde al mirror de Brasil con HTTP, luego se guardan estos cambios.

```
# Here are some individual mirrors that can be used instead of the
# redirector at mirrors.slackware.com if necessary ; note that this
# list is not guaranteed to be up-to-date
#
# AUSTRALIA (AU)
# ftp://ftp.cc.swin.edu.au/slackware/slackware64-15.0/
# http://ftp.cc.swin.edu.au/slackware/slackware64-15.0/
# ftp://ftp.iinet.net.au/pub/slackware/slackware64-15.0/
# http://ftp.iinet.net.au/pub/slackware/slackware64-15.0/
# ftp://mirror.as24220.net/pub/slackware/slackware64-15.0/
# http://mirror.as24220.net/pub/slackware/slackware64-15.0/
# ftp://mirror.internode.on.net/pub2/slackware/slackware64-15.0/
# http://mirror.internode.on.net/pub/slackware/slackware64-15.0/
# AUSTRIA (AT)
# http://gd.tuwien.ac.at/opsys/linux/freesoftware.com/slackware64-15.0/
# BELARUS (BY)
# ftp://mirror.datacenter.by/pub/slackware/slackware64-15.0/
# http://mirror.datacenter.by/pub/slackware/slackware64-15.0/
# BRAZIL (BR)
# ftp://ftp.slackware-brasil.com.br/slackware64-15.0/
# http://ftp.slackware-brasil.com.br/slackware64-15.0/
# BULGARIA (BG)
# ftp://mirrors.unixsol.org/slackware/slackware64-15.0/
#
# Uncomment one mirror only.
#
/etc/slackpkg/slackpkg.conf: 157 lines, 6111 characters.
root@darkstar:~# slackpkg install samba
```

Ahora si procedemos con la instalación de Samba, se ejecuta el comando `slackpkg install samba`, aparece una pantalla azul y se presiona ok para que inicie la instalación de Samba.

	<pre>slackpkg 15.0.10</pre>  <pre> => SIZE samba-4.18.9-i586-1_slack15.0.tgz ... 13757428 => PASU ... done. => RETR samba-4.18.9-i586-1_slack15.0.tgz ... done. length: 13757428 (13M) (unauthoritative) samba-4.18.9-i586-1_slac 100%(=====)l 13.12M 1.74MB/s in 16s 2025-09-01 12:25:20 (846 KB/s) - '//var/cache/packages/./patches/packages/samba-4.18.9-i586-1_slack1 s.0.tgz' saved [13757428] Package samba-4.18.9-i586-1_slack15.0.tgz is already in cache - not downloading Installing samba-4.18.9-i586-1_slack15.0... Verifying package samba-4.18.9-i586-1_slack15.0.tgz: Installing package samba-4.18.9-i586-1_slack15.0.tgz: PACKAGE DESCRIPTION: samba (CIFS file and print server) Samba is a CIFS file and print server for CIFS clients. It allows you to make file space or printers on a Samba host available to CIFS clients (such as PCs running Windows). If you have any Windows file servers, you may be able to replace them or supplement them with Samba. One of Samba's big strengths is integration, so you can use it to tie together your Linux hosts and Windows PC clients. Executing install script for samba-4.18.9-i586-1_slack15.0.tgz. Package samba-4.18.9-i586-1_slack15.0.tgz installed. Searching for NEW configuration files... No .new files found. root@darkstar:~# </pre>
<p>Una vez instalado Samba, para poder usar el usuario registrado en Solaris, se debe instalar la librería cifs-utils ya que es la que nos permite montar carpetas compartidas desde otra maquina en la red y libcap-ng garantiza que Samba y sus utilidades manejen permisos sin romper la seguridad del sistema.</p>	<pre>root@darkstar:~# slackpkg install libcap-ng_</pre>  <pre>slackpkg 15.0.10</pre>

	
Al tener ya instaladas estas librerías, se procede a montar el recurso compartido que se encuentra ubicado en //10.2.77.180/shared y se va a almacenar en /mnt/solaris_share, usando las credenciales del usuario aurora.	<pre>root@darkstar:~# mount -t cifs //10.2.77.180/shared /mnt/solaris_share -o user=aurora Password for aurora@//10.2.77.180/shared: root@darkstar:~#</pre>
Ahora se ejecuta el comando ls /mnt/solaris_share y aparece el .txt que fue añadido anteriormente desde la máquina de Windows.	<pre>root@darkstar:~# ls /mnt/solaris_share Himno.txt* root@darkstar:~# _</pre>

Conociendo Cloud

En el presente se documenta el proceso de implementación y conexión de una instancia virtual mediante Amazon Elastic Compute Cloud (Amazon EC2), como parte de las prácticas realizadas en el entorno de laboratorio proporcionado por AWS Academy. El objetivo principal de esta actividad es comprender de manera práctica el funcionamiento de EC2, uno de los servicios fundamentales de Amazon Web Services (AWS) dentro del paradigma de computación en la nube.

Cada integrante del equipo llevó a cabo de forma individual los mismos pasos descritos en este informe, con el fin de asegurar una comprensión profunda del tema y fomentar el aprendizaje autónomo. Esta metodología permitió que todos los participantes experimentaran directamente el ciclo completo de despliegue, configuración y conexión a una instancia EC2, reforzando así los conceptos teóricos abordados previamente.

Esta sección corresponde a la implementación realizada por Santiago, quien siguió paso a paso el procedimiento de lanzamiento y conexión de una instancia EC2 dentro del entorno de laboratorio de AWS Academy. Esta experiencia le permitió afianzar los conceptos fundamentales de la computación en la nube y el uso de Amazon EC2 como servicio de infraestructura escalable.

Actividad/Acción/Tarea	Detalles relevantes																				
Se utilizará el entorno de laboratorio proporcionado por el curso de AWS Academy Learner con el objetivo de acceder al Manager Center. Para ello, es necesario iniciar dicho entorno, el cual incluye un crédito gratuito de \$50. Una vez activado, se podrá proceder con la apertura y utilización del Manager Center para realizar las actividades correspondientes.																					
En la barra de navegación ubicada en la parte superior de la consola de Amazon EC2 se muestra la región de AWS seleccionada por defecto, la cual puede ser modificada según la ubicación geográfica del usuario para optimizar el rendimiento. Para determinar la región más adecuada, se utilizó la herramienta AWS Latency Test , mediante la cual se identificó que la región us-east-1 (N. Virginia) presenta la mejor latencia desde nuestra ubicación. Por lo tanto, se seleccionó esta región para llevar a cabo la implementación de la instancia EC2, asegurando así una conexión más eficiente y estable.	 <table><thead><tr><th>GEOGRAPHY</th><th>REGION</th><th>LOCATION</th><th>PERFORMANCE</th><th>LATENCY (ms)</th></tr></thead><tbody><tr><td>Americas</td><td>us-east-1</td><td>N. Virginia</td><td>ACCEPTABLE</td><td>187 ms</td></tr><tr><td>Americas</td><td>us-east-2</td><td>Ohio</td><td>ACCEPTABLE</td><td>171 ms</td></tr><tr><td>Americas</td><td>ca-central-1</td><td>Central</td><td>ACCEPTABLE</td><td>116 ms</td></tr></tbody></table>	GEOGRAPHY	REGION	LOCATION	PERFORMANCE	LATENCY (ms)	Americas	us-east-1	N. Virginia	ACCEPTABLE	187 ms	Americas	us-east-2	Ohio	ACCEPTABLE	171 ms	Americas	ca-central-1	Central	ACCEPTABLE	116 ms
GEOGRAPHY	REGION	LOCATION	PERFORMANCE	LATENCY (ms)																	
Americas	us-east-1	N. Virginia	ACCEPTABLE	187 ms																	
Americas	us-east-2	Ohio	ACCEPTABLE	171 ms																	
Americas	ca-central-1	Central	ACCEPTABLE	116 ms																	
Una vez dentro del dashboard de Amazon EC2, se debe hacer clic en el botón "Launch Instance" para iniciar el proceso de creación de una nueva máquina virtual en la nube. Este procedimiento permite configurar los parámetros necesarios para desplegar una instancia que se ajusta a los requerimientos del entorno de laboratorio.																					

Durante el proceso de creación de la instancia, se asignó un nombre descriptivo: AWS-Cloud-Development. En la sección Application and OS Images (Amazon Machine Image), se seleccionó como sistema operativo Amazon Linux, específicamente la imagen Amazon Linux 2023 con kernel 6.1 AMI, la cual es elegible para la capa gratuita (Free Tier). Esta elección permite realizar pruebas y configuraciones sin incurrir en costos adicionales.

En el paso correspondiente a la elección del tipo de instancia, se seleccionó t3.micro, una opción compatible con la capa gratuita (Free Tier) de AWS. Esta instancia ofrece una configuración equilibrada de recursos que resulta adecuada para entornos de desarrollo y pruebas, permitiendo optimizar el uso de los créditos disponibles sin generar costos adicionales.

Se creó un par de claves con un nombre descriptivo: my-key-pair, con el fin de facilitar su identificación y reutilización en futuras conexiones. En cuanto a la configuración técnica:

- Tipo de clave (Key pair type): Se seleccionó RSA, por su amplia compatibilidad con la mayoría de los clientes SSH.
- Formato de la clave privada (Private key format): Se eligió el formato .pem, ideal para su uso con OpenSSH en sistemas Linux/macOS. Este formato también permite la conexión desde PowerShell en Windows mediante SSH.

Esta configuración será utilizada posteriormente para establecer la conexión entre máquinas virtuales, incluyendo una instancia con Slackware.

Se mantuvieron las opciones predeterminadas para simplificar el despliegue en el entorno de laboratorio. En el apartado de firewall, se seleccionó la opción "Create new security group", con una regla que permite el tráfico SSH (puerto 22) desde cualquier fuente.

- Tipo de tráfico permitido: SSH
- Origen: 0.0.0.0/0 (acceso abierto desde cualquier dirección IP)

Además, la instancia fue configurada para asignar automáticamente una dirección IP pública, lo que permite establecer conexiones remotas sin necesidad de configurar una Elastic IP manualmente.

Se configuró un único volumen raíz con las siguientes características:

- Tipo de volumen: gp3
- Capacidad: 8 GiB
- Encriptación: No se aplicó encriptación
- Propósito: Volumen raíz del sistema

Esta configuración fue seleccionada por ser suficiente para las necesidades del entorno de pruebas, permitiendo un rendimiento adecuado sin exceder los límites de la capa gratuita (Free Tier).

Con todas las configuraciones previamente establecidas, el entorno está listo para proceder con la creación de la instancia. En el panel de resumen, se seleccionó una única instancia y se hizo clic en el botón “Launch Instance” para iniciar el proceso de despliegue.

A partir de este momento, es necesario esperar a que el estado de la instancia cambie a “running”, lo que indica que el despliegue se ha completado exitosamente y la máquina virtual está lista para ser utilizada.

▼ Summary

Number of instances [Info](#)

1

Software Image (AMI)
Amazon Linux 2023 AMI 2023.8.2...[read more](#)
ami-00ca32bbc84273381

Virtual server type (instance type)
t3.micro

Firewall (security group)
New security group

Storage (volumes)
1 volume(s) - 8 GiB

[Cancel](#) [Launch instance](#) [Preview code](#)

Una vez que la instancia se encuentra en estado “running”, es fundamental verificar que los indicadores de salud estén activos y sin errores. Para ello, se debe revisar el panel de “Status Checks and Alarms”, donde deben aparecer los siguientes controles:

- **System Status Check:** Verifica que la infraestructura subyacente de AWS (hardware, red, etc.) esté funcionando correctamente.
- **Instance Status Check:** Evalúa que el sistema operativo dentro de la instancia esté respondiendo de manera adecuada.

Ambos indicadores deben mostrar el estado “passed” para confirmar que la instancia está operativa y lista para ser utilizada en el entorno de pruebas.

Details **Status and alarms** Monitoring Security Networking Storage Tags

Status checks [Info](#) [Actions](#)

Status checks detect problems that may impair i-0a64182ae3b214441 (AWS-Cloud-Deployment) from running your applications.

System status checks	Instance status checks	Attached EBS status checks
✔ System reachability check passed	✔ Instance reachability check passed	✔ Attached EBS reachability check passed

► Metrics

▼ Alarms

Find alarms by name

Name	State	Description	Metric name	State reason
Instance has no associated alarms				

Para verificar el funcionamiento correcto de la instancia, se debe hacer clic en el botón “Connect” dentro del panel de EC2. Esto abrirá una ventana con las opciones de conexión disponibles.

Para establecer una conexión estable y segura, se selecciona la opción “SSH client”, la cual proporciona un comando de ejemplo que incluye la IP pública de la instancia y el nombre del archivo de la clave privada. Este comando debe copiarse desde la sección “Example” y ejecutarse en una terminal compatible con SSH (como PowerShell en Windows o Terminal en Linux/macOS).

Una vez configurado el Key Pair durante la creación de la instancia, se descargó automáticamente un archivo en el formato seleccionado (.pem en este caso). Este archivo contiene la clave privada necesaria para autenticar la conexión SSH.

Para conectarse a la instancia:

- Abrir una terminal en el directorio donde se encuentra el archivo .pem.
- Pegar el comando SSH proporcionado en el panel de conexión de EC2 (sección "Example" bajo la opción "SSH client").
- Al ejecutar el comando por primera vez, el sistema solicitará confirmar la autenticidad del host. Se debe escribir "yes" para continuar.

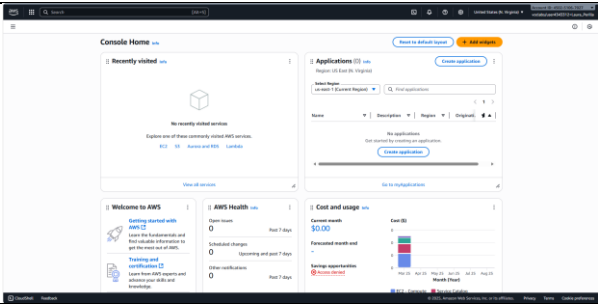
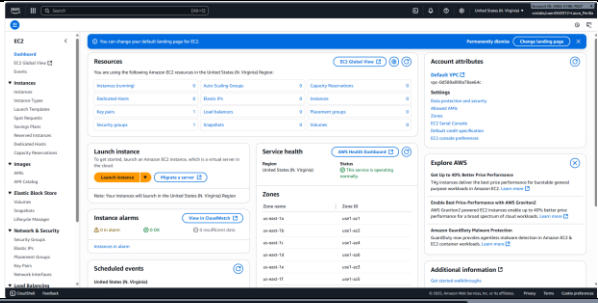
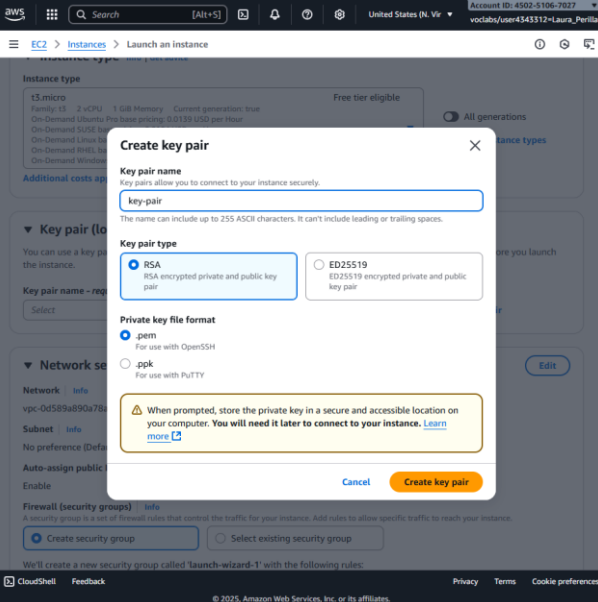
Si todo está correctamente configurado, se establecerá una sesión remota en la consola de la instancia EC2, permitiendo interactuar directamente con el sistema operativo desde la terminal.

The image shows two screenshots related to connecting to an AWS EC2 instance via SSH.

The top screenshot is from the AWS Management Console, specifically the 'Connect' page for an EC2 instance. It shows the 'SSH client' tab selected. The 'Instance ID' is 'i-0a64182ae3b214441' (AWS-Cloud-Deployment). The steps listed are: 1. Open an SSH client. 2. Locate your private key file. 3. Run the command, if necessary, to ensure your key is not publicly viewable. 4. Connect to your instance using its Public DNS. An example command is provided: `ssh -i "my-key-pair.pem" ec2-user@ec2-34-207-239-251.compute-1.amazonaws.com`. A note states: "Note: In most cases, the guessed username is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username."

The bottom screenshot is a Windows PowerShell terminal window. It shows the execution of the SSH command: `PS C:\Users\Redes\Downloads> ssh -i "my-key-pair.pem" ec2-user@ec2-34-207-239-251.compute-1.amazonaws.com`. The terminal output shows the host key fingerprint and a warning to permanently add the host to the list of known hosts. The user responds with 'yes'. The terminal then shows the connection to the EC2 instance, displaying the Amazon Linux 2023 logo and the URL `https://aws.amazon.com/linux/amazon-linux-2023`. The prompt changes to `[ec2-user@ip-172-31-39-147 ~]$`.

Esta sección documenta la implementación llevada a cabo por Natalia, quien replicó de forma individual el proceso de despliegue de una instancia EC2 en el laboratorio de AWS Academy. Natalia realizó la configuración completa de la instancia, desde la selección de parámetros iniciales hasta la conexión remota mediante SSH, permitió reforzar su conocimiento sobre la gestión de recursos en la nube y el funcionamiento de EC2 como herramienta clave en entornos virtualizados.

Actividad/Acción/Tarea	Detalles relevantes
Se inicia sesión con mi cuenta de AWS y se selecciona la región, se dejó la predeterminada porque es la región con mejor latencia	
En el panel principal de EC2, busca la sección Launch instance y haz clic en Launch instance.	
Se le asigna un nombre a la instancia, en mi caso la nombre Cloud-Linux-Shell, en el resto de opciones se dejan las que vienen predeterminadas y se procede a crear una Key Pair.	

Después de que ambos implementaran máquinas con Amazon Linux como parte del proceso de aprendizaje, ahora se procederá a realizar la instalación del sistema operativo Slackware en una instancia EC2 en la nube. Esta actividad busca ampliar el conocimiento sobre la administración y configuración de sistemas menos convencionales en entornos virtualizados de AWS, explorando la compatibilidad y adaptabilidad de distintas distribuciones Linux en la infraestructura cloud.

Sin embargo, se identificó que Slackware no es compatible directamente con la imagen base Amazon Linux proporcionada por AWS, lo que impide su instalación mediante las imágenes estándar disponibles en el Marketplace o el catálogo de AMIs públicas. Debido a esta limitación, será necesario crear una Amazon Machine Image (AMI) personalizada que permita desplegar Slackware en una instancia EC2, asegurando que el sistema operativo funcione correctamente y pueda aprovechar los recursos del entorno de AWS.

La creación de esta AMI personalizada implica configurar manualmente una instancia base con Slackware instalado y optimizado para el entorno de Amazon EC2, para luego generar una imagen que pueda ser reutilizada en futuros despliegues. Este proceso permitirá además familiarizarse con tareas avanzadas de personalización y administración de imágenes en la nube, esenciales para entornos donde se requieren configuraciones específicas que no están cubiertas por las opciones predeterminadas.

De esta manera, se garantiza que la instalación y uso de Slackware en AWS EC2 se realice de forma eficiente y segura, superando las limitaciones iniciales y ampliando las capacidades del equipo en el manejo de infraestructuras basadas en la nube. Este enfoque práctico refuerza el aprendizaje sobre la flexibilidad de AWS y la importancia de adaptar los recursos a las necesidades particulares de cada proyecto o entorno.

Actividad/Acción/Tarea	Detalles relevantes
Una vez se lanzó una instancia EC2 en AWS con Amazon Linux como sistema operativo.	
Desde PowerShell en Windows, se utilizó el comando scp para copiar el script buscar_archivo.sh desde el equipo local a la instancia EC2.	
Una vez dentro de la instancia, se le pueden dar permisos de ejecución al script y ya con eso, se puede ejecutar el script.	
Se creo una carpeta y un archivo txt para probar el correcto funcionamiento del script, en la imagen se observa que el script esta funcionando correctamente.	

Resultados

En este apartado se presentan los hallazgos obtenidos durante el desarrollo del laboratorio. A lo largo de las actividades, se realizaron diversas pruebas y simulaciones empleando herramientas como Packet Tracer, Wireshark y entornos virtualizados, con el objetivo de observar y comprender el comportamiento de sistemas operativos, dispositivos de red, y protocolos de comunicación. Los resultados aquí descritos reflejan tanto el proceso de configuración como la interpretación de datos generados, permitiendo validar el funcionamiento de las soluciones implementadas y el cumplimiento de los objetivos propuestos en la guía de laboratorio.

Experimentos

¿Qué versión de Packet Tracer está disponible en la plataforma Cisco?

Actualmente, la última versión estable de Cisco Packet Tracer disponible en la plataforma oficial de Cisco es la versión 8.2.2. Esta versión se encuentra disponible para descarga gratuita a través del portal Cisco Networking Academy (NetAcad), aunque es necesario estar registrado como estudiante o instructor en la plataforma para acceder a ella. (Cisco Networking Academy, n.d.)

¿Qué representan las conexiones negras sólidas?

Los cables de conexión directa son el tipo más común de cable Ethernet utilizado en redes. Su principal función es conectar dispositivos que operan en diferentes capas del modelo de red, como por ejemplo un computador con un switch, o un switch con un router.

En este tipo de cable, los hilos internos están organizados en el mismo orden en ambos extremos, lo que permite que los pines de transmisión (TX) de un dispositivo se conecten directamente con los pines de recepción (RX) del otro, y viceversa.

El esquema de cableado estándar para los cables de conexión directa se denomina T568A, y sigue el siguiente orden de colores de izquierda a derecha:

Blanco-verde, verde, blanco-naranja, azul, blanco-azul, naranja, blanco-marrón, marrón.

Este tipo de cableado garantiza una comunicación eficiente entre dispositivos de red que no comparten la misma función dentro de la arquitectura del sistema. (Mack, 2025)

¿Qué representan las conexiones discontinuas de color negro?

Los cables cruzados se utilizan para conectar dispositivos que operan en la misma capa del modelo de red, como dos computadores, dos switches o dos routers. A diferencia de los cables de conexión directa, en los cables cruzados los hilos internos están organizados en un orden distinto en cada extremo del cable.

Esto significa que los pines de transmisión (TX) de un dispositivo se conectan directamente con los pines de transmisión del otro, y lo mismo ocurre con los pines de recepción (RX). Esta configuración permite la comunicación directa entre dispositivos similares sin necesidad de un intermediario como un switch.

El esquema de cableado estándar para los cables cruzados es una combinación de T568A en un extremo y T568B en el otro. El orden de los hilos en el extremo T568A, de izquierda a derecha, es el siguiente:

Blanco-verde, verde, blanco-naranja, azul, blanco-azul, naranja, blanco-marrón, marrón.

Este tipo de cableado es útil en entornos de prueba o cuando se requiere una conexión directa entre dispositivos de red equivalentes. (Mack, 2025)

En Red Real

¿Qué es Wireshark?

Wireshark es una herramienta especializada en el análisis de paquetes de red, diseñada para ofrecer una visualización detallada de los datos capturados durante la transmisión de información en redes informáticas. A diferencia de los analizadores de paquetes tradicionales, que solían ser costosos y de acceso limitado, Wireshark representa una solución moderna, gratuita y de código abierto, ampliamente reconocida por su eficacia y versatilidad.

Las aplicaciones de Wireshark abarcan múltiples áreas dentro del ámbito tecnológico. Es utilizado por administradores de red para identificar y resolver problemas de conectividad, por ingenieros de seguridad para investigar incidentes relacionados con la protección de datos, y por profesionales de aseguramiento de calidad para verificar el comportamiento de aplicaciones que operan sobre redes. Asimismo, los desarrolladores lo emplean para depurar implementaciones de protocolos, y los entusiastas de la informática lo consideran una herramienta educativa para

comprender el funcionamiento interno de los protocolos de comunicación. Su utilidad se extiende a diversos contextos técnicos, formativos y de investigación.

Entre sus principales características, Wireshark ofrece compatibilidad con sistemas operativos UNIX y Windows, y permite la captura de datos en tiempo real desde diversas interfaces de red. Es capaz de abrir archivos generados por otros programas de captura como tcpdump y WinDump, así como importar volcados hexadecimales desde archivos de texto. Los paquetes capturados se presentan con información detallada sobre los protocolos involucrados, y pueden ser almacenados o exportados en múltiples formatos. Además, Wireshark incluye funciones avanzadas de filtrado y búsqueda, opciones de visualización mediante colores personalizados, y herramientas estadísticas que enriquecen el análisis de tráfico. Estas funcionalidades convierten a Wireshark en una solución integral para el estudio de redes.

La herramienta también destaca por su capacidad para capturar tráfico desde una amplia gama de medios de red, incluyendo Ethernet, redes inalámbricas, Bluetooth y USB. No obstante, la compatibilidad con ciertos medios puede estar condicionada por el hardware y el sistema operativo del equipo utilizado. Para obtener información detallada sobre los medios soportados, se recomienda consultar la documentación oficial disponible en el sitio web del proyecto. En cuanto a la interoperabilidad, Wireshark permite importar y exportar archivos de captura en formatos compatibles con otros programas, lo que facilita su integración en distintos entornos de trabajo. (*Chapter 1. Introduction*, s. f.)

¿Qué significa que una tarjeta de red esté en modo promiscuo?

El modo promiscuo en redes informáticas permite que un dispositivo de red, como una tarjeta de interfaz de red (NIC) o un adaptador en un sistema anfitrión, intercepte y lea todos los

paquetes que circulan por la red, sin limitarse a aquellos dirigidos específicamente a él. En condiciones normales, las NIC filtran el tráfico para recibir únicamente los paquetes destinados al dispositivo, pero al habilitar el modo promiscuo, esta restricción se elimina, permitiendo el acceso completo al flujo de datos en la red. Esta funcionalidad es especialmente útil en tareas de seguridad, monitoreo y administración, ya que permite visualizar todo el tráfico, facilitando la detección de intrusiones, el diagnóstico de fallos y el análisis de actividad. No obstante, también implica riesgos, ya que puede ser explotada por agentes maliciosos para interceptar comunicaciones, comprometer sistemas o sustraer información sensible.

El modo promiscuo es aplicable tanto a interfaces cableadas como inalámbricas, y al activarse, garantiza que todos los paquetes transmitidos sean procesados por el controlador y enviados directamente a la CPU sin aplicar filtros. Esto permite que el sistema operativo o herramientas de monitoreo accedan al contenido de los paquetes, lo cual es útil para identificar amenazas o problemas de rendimiento como la latencia. En redes puenteadas, puede ser necesario que la NIC opere en modo promiscuo para asegurar una transmisión adecuada de datos, siendo indispensable que tanto el adaptador como el controlador del sistema anfitrión sean compatibles. Algunos sistemas operativos requieren privilegios de superusuario para habilitar esta función, y es posible configurarla para que los datos sean accesibles a sistemas operativos invitados dentro del entorno anfitrión.

Una de las aplicaciones más comunes del modo promiscuo es el análisis de paquetes o sniffing, que consiste en recolectar y registrar los paquetes que circulan por la red para su análisis posterior, con fines como estudiar el tráfico, evaluar el uso del ancho de banda o detectar anomalías. Aunque cualquier NIC puede recibir todo el tráfico, la mayoría están configuradas para ignorar paquetes no dirigidos a ellas; los sniffers modifican esta configuración mediante

comandos como `ifconfig` o `ip link set`, o mediante herramientas especializadas. Una vez activado, el sniffer recolecta todo el tráfico de la interfaz física, lo organiza y lo registra según los requerimientos de la red, permitiendo que todo el tráfico sea procesado por la pila de red sin importar su destino. La mayoría de los sniffers actuales son de software, aunque existen versiones en hardware. `Tcpdump` es un ejemplo destacado, ya que permite visualizar paquetes TCP/IP desde la línea de comandos y guardar los datos para análisis posterior, como la resolución de problemas de conectividad o la revisión de consultas DNS.

Wireshark es otra herramienta ampliamente utilizada, de código abierto, que captura y presenta datos de paquetes en tiempo real. Su funcionalidad incluye la solución de problemas de red, la identificación de brechas de seguridad, la verificación de aplicaciones y la depuración de protocolos. Disponible para Windows y Linux, permite filtrar, buscar, guardar y exportar paquetes en distintos formatos, lo que la convierte en una herramienta versátil para administradores y analistas. El modo promiscuo también se emplea para monitorear la actividad de red y diagnosticar problemas de conectividad, asignándose en algunos casos a servidores espía que capturan y almacenan todos los paquetes para su análisis, permitiendo estudiar el comportamiento de la red y tomar decisiones de optimización. Asimismo, los sistemas de detección y prevención de intrusiones lo utilizan para identificar actividades sospechosas o maliciosas. (Awati, 2025)

Software Base

¿Qué son los archivos de log?

Los archivos de log (o registros) son archivos de texto donde el sistema operativo y las aplicaciones escriben eventos importantes, como errores del sistema, intentos de inicio de sesión, actividad del kernel y servicios iniciados o detenidos.

¿Qué tipos de logs hay en los sistemas operativos que instalaste?

Slackware

- /var/log/messages, /var/log/secure, /var/log/dmesg, /var/log/cron, /var/log/boot.log

Solaris

- /var/adm/messages, /var/log/syslog, /var/adm/loginlog

Windows

- Logs del Visor de Eventos: Aplicación, Seguridad, Sistema

Android

- /data/log, /sys/fs/pstore/, o en Almacenamiento interno/logback

¿Qué es syslog? ¿Qué define este estándar? ¿Los logs que encontraste en los sistemas operativos siguen este estándar?

Syslog es un estándar de registro de eventos usado en sistemas Unix/Linux. Este define:

- Formato del mensaje: prioridad, timestamp, hostname, proceso, mensaje.
- Niveles de severidad: desde emerg (0) hasta debug (7).

- Facilidades: origen del mensaje (kernel, mail, auth, etc.).
- Transporte: puede enviar logs a archivos locales o a servidores remotos.

Sí, los sistemas Slackware y Solaris siguen el estándar Syslog o utilizan variantes modernas compatibles con él, Android no sigue el estándar directamente y Windows tampoco usa Syslog como estándar nativo.

¿Qué comando se puede usar para eliminar una palabra en VI?

En VI existen varias formas de eliminar palabras, según lo que se necesite:

- dw: elimina la palabra desde la posición actual del cursor hasta el final de la palabra.
- daw: elimina la palabra completa donde está el cursor, incluyendo el espacio después.
- diw: elimina la palabra completa pero sin tocar los espacios en blanco alrededor.

Conociendo Cloud

¿Cuáles son las principales características de Amazon EC2?

Amazon Elastic Compute Cloud (EC2) ofrece un conjunto de funcionalidades que optimizan el rendimiento, la flexibilidad y la escalabilidad de las aplicaciones en la nube. A continuación, se describen las características clave:

- 1. Múltiples Ubicaciones Geográficas:** Amazon EC2 permite desplegar instancias en múltiples ubicaciones. Estas ubicaciones están compuestas por Regiones y Zonas de Disponibilidad. Las Zonas de Disponibilidad son ubicaciones independientes diseñadas para estar aisladas de fallos en otras zonas, lo que proporciona conectividad de red de baja latencia y de bajo costo entre ellas. El despliegue de instancias en diferentes zonas ayuda a proteger las aplicaciones de fallos en un solo punto. Cada Región de EC2 se compone de una o más Zonas de Disponibilidad y está distribuida geográficamente, con un compromiso de disponibilidad del 99,99% por parte de Amazon EC2.
- 2. Tiempo Preciso con el Servicio de Sincronización de Hora de Amazon:** Amazon EC2 utiliza el Servicio de Sincronización de Hora de Amazon, que proporciona una fuente de tiempo precisa y confiable para los servicios de AWS, incluidas las instancias EC2. Este servicio asegura la consistencia en la sincronización de hora entre las instancias y otros servicios de la nube.
- 3. Elección de Sistemas Operativos y Software:** Amazon EC2 ofrece una amplia variedad de Imágenes de Máquina de Amazon (AMIs) preconfiguradas, que incluyen sistemas operativos como Microsoft Windows y distribuciones de Linux (Amazon Linux 2, Ubuntu, Red Hat Enterprise Linux, CentOS, SUSE, Debian). Además, el AWS

Marketplace proporciona acceso a una amplia selección de software comercial y gratuito de proveedores reconocidos, diseñado específicamente para ejecutar en instancias EC2.

4. **Pago por Uso:** Con el modelo de facturación por segundo, solo se paga por lo que se utiliza, eliminando los costos de minutos y segundos no utilizados. Esta opción permite a los usuarios centrarse en mejorar sus aplicaciones sin necesidad de maximizar el uso hasta completar la hora.
5. **Escalado Automático con Amazon EC2 Auto Scaling:** Amazon EC2 Auto Scaling permite ajustar automáticamente la capacidad de las instancias EC2 según las condiciones definidas por el usuario. A través de políticas de escalado dinámico y predictivo, es posible añadir o eliminar instancias EC2 de manera eficiente, garantizando un rendimiento constante durante picos de demanda y reduciendo costos durante periodos de baja demanda.
6. **Optimización de Rendimiento y Costos con Amazon EC2 Fleet:** Amazon EC2 Fleet permite provisionar capacidad de cómputo a través de diferentes tipos de instancias, Zonas de Disponibilidad y modelos de compra con un solo llamado a la API. Esta funcionalidad facilita la optimización del rendimiento y los costos al adaptar la infraestructura de manera dinámica.
7. **Configuraciones de CPU Optimizada:** La característica Optimize CPUs otorga un control adicional sobre las instancias EC2. Los usuarios pueden especificar el número de vCPUs o desactivar la Tecnología Intel Hyper-Threading (Intel HT) para aplicaciones que se benefician de CPUs de un solo hilo, como algunas aplicaciones de alto rendimiento (HPC).

8. **Pausar y Reanudar Instancias:** Amazon EC2 permite hibernar instancias respaldadas por Amazon EBS y reanudarlas en un momento posterior. Esta funcionalidad es útil para aplicaciones que requieren un tiempo significativo para arrancar y mantener su estado en la memoria.
9. **Almacenamiento Óptimo para Cada Carga de Trabajo:** EC2 ofrece varias opciones de almacenamiento para adaptarse a diferentes tipos de cargas de trabajo. Amazon EBS proporciona almacenamiento de bloques persistente y de baja latencia, mientras que Amazon EFS ofrece almacenamiento de archivos gestionado y escalable, ideal para accesos compartidos.
10. **Redes Mejoradas para Alto Rendimiento:** La opción de Redes Mejoradas mejora el rendimiento de paquetes por segundo (PPS), reduce la latencia y mejora la eficiencia de la red. Esta característica permite un rendimiento de I/O superior y menor utilización de CPU en comparación con implementaciones tradicionales.
11. **Intercomunicación de Alto Nivel con Elastic Fabric Adapter (EFA):** EFA es una interfaz de red para instancias EC2 que permite ejecutar aplicaciones que requieren alta intercomunicación entre instancias, como simulaciones de dinámica de fluidos o modelado de predicción meteorológica, aprovechando la baja latencia y el alto rendimiento de red.
12. **Direcciones IP Elásticas para la Gestión Dinámica de la Nube:** Las direcciones IP elásticas proporcionan direcciones IP estáticas que están asociadas a una cuenta, no a una instancia en particular, lo que facilita la remapeo de direcciones IP públicas a cualquier instancia dentro de la cuenta, permitiendo una recuperación rápida ante fallos de instancias o Zonas de Disponibilidad.

13. Cálculo de Alto Rendimiento con Clústeres HPC: Amazon EC2 soporta clústeres de Cálculo de Alto Rendimiento (HPC), que permiten realizar procesos paralelizados de alto rendimiento en aplicaciones sensibles a la red. Instancias como Cluster Compute, Cluster GPU y High Memory Cluster están diseñadas para ofrecer alto rendimiento de red y escalabilidad para aplicaciones intensivas en recursos.

14. Acceso Seguro a Servicios con AWS PrivateLink: AWS PrivateLink es una tecnología que permite a los usuarios acceder a servicios de Amazon de manera segura y de alto rendimiento, manteniendo todo el tráfico de red dentro de la infraestructura de AWS.

15. Mantenimiento Sin Interrupciones: Amazon EC2 emplea tecnologías como actualizaciones en vivo y migración en vivo para realizar mantenimiento sin interrumpir las instancias. Estas tecnologías permiten desplegar parches de seguridad, nuevas características o mejoras de rendimiento sin necesidad de reiniciar las instancias, lo que mejora el tiempo de actividad y reduce el esfuerzo operativo.

(Features, s. f.)

¿Qué servicios se pueden utilizar con Amazon EC2?

Amazon EC2 se integra con una variedad de servicios de AWS que permiten extender su funcionalidad y mejorar la gestión, seguridad y escalabilidad de sus instancias. A continuación, se describen los principales servicios complementarios:

- **Amazon EC2 Auto Scaling:** Permite asegurar que siempre haya la cantidad adecuada de instancias EC2 disponibles para manejar la carga de su aplicación, ajustándose automáticamente según la demanda.

- **AWS Backup:** Automatiza el respaldo de instancias EC2 y de los volúmenes de Amazon EBS asociados, facilitando la protección de datos.
- **Amazon CloudWatch:** Ofrece monitoreo en tiempo real de las instancias EC2 y los volúmenes de EBS, permitiendo recopilar métricas, generar alarmas y visualizar logs.
- **Elastic Load Balancing (ELB):** Distribuye automáticamente el tráfico de aplicaciones entrante entre múltiples instancias EC2, mejorando la disponibilidad y tolerancia a fallos.
- **Amazon GuardDuty:** Servicio de detección de amenazas que identifica el uso no autorizado o malicioso de las instancias EC2 mediante análisis continuo de seguridad.
- **EC2 Image Builder:** Automatiza la creación, gestión y despliegue de imágenes de servidor personalizadas, seguras y actualizadas.
- **AWS Launch Wizard:** Facilita el dimensionamiento, configuración y despliegue de recursos de AWS para aplicaciones de terceros, sin necesidad de aprovisionar manualmente cada recurso.
- **AWS Systems Manager:** Proporciona una solución integral para la administración segura de instancias EC2 a gran escala, permitiendo ejecutar tareas operativas automatizadas.

Además de Amazon EC2, AWS ofrece otros servicios de cómputo que pueden utilizarse según las necesidades del proyecto:

- **Amazon Lightsail:** Plataforma de nube simplificada ideal para crear sitios web o aplicaciones web, que ofrece recursos preconfigurados con precios mensuales predecibles y bajos. Es una buena opción para proyectos más pequeños o con requerimientos menos complejos.

- **Amazon Elastic Container Service (Amazon ECS):** Permite implementar, administrar y escalar aplicaciones en contenedores sobre un clúster de instancias EC2. Es adecuado para cargas de trabajo basadas en Docker.
- **Amazon Elastic Kubernetes Service (Amazon EKS):** Permite ejecutar aplicaciones basadas en Kubernetes en la nube de AWS, brindando una solución administrada para orquestación de contenedores.

(What Is Amazon EC2? - Amazon Elastic Compute Cloud, s. f.)

¿Cuál es la diferencia entre detener, finalizar y reiniciar una instancia EC2?

Amazon EC2 permite administrar el estado de las instancias mediante tres acciones principales: terminar, detener y reiniciar. Cada una tiene implicaciones distintas en cuanto a costos, disponibilidad de datos y comportamiento del sistema. Es fundamental entender estas diferencias para gestionar eficientemente los recursos en la nube.

- **Terminar una instancia (Terminate Instance)**
 - Qué ocurre: La instancia se apaga y la máquina virtual asignada se libera de forma permanente.
 - Costos: Se deja de facturar por el uso de la instancia.
 - Datos: Toda la información almacenada en el disco local (almacenamiento efímero) se pierde.
 - Los volúmenes EBS adjuntos también se eliminan por defecto, a menos que se configure lo contrario.
 - Uso recomendado: Cuando ya no se necesita la instancia y se desea evitar cualquier cargo adicional.

- **Detener una instancia (Stop Instance)**

- Qué ocurre: La instancia se apaga, pero no se elimina. El volumen EBS raíz se conserva y la instancia puede reiniciarse posteriormente.
- Costos: No se cobra por el uso de la instancia mientras está detenida. Sí se cobra por el almacenamiento de los volúmenes EBS asociados.
- Datos: El almacenamiento local se pierde. El volumen EBS persiste y mantiene su información.
- IP y DNS: Al reiniciarla, la instancia recibe una nueva dirección IP pública y DNS, aunque conserva el mismo ID de instancia.
- Uso recomendado: Cuando se planea volver a utilizar la instancia en el corto plazo y se desea conservar su configuración y datos.
- Importante: Solo se pueden detener instancias que fueron lanzadas utilizando una AMI basada en EBS.

- **Reiniciar una instancia (Reboot Instance)**

- Qué ocurre: Se realiza un reinicio del sistema operativo, similar al de un ordenador personal.
- Costos: La instancia continúa en estado "en ejecución" y se siguen generando cargos.
- Datos: Todos los datos persisten, tanto en el almacenamiento local como en los volúmenes EBS.
- IP y DNS: No cambian. La instancia continúa utilizando las mismas direcciones.
- Uso recomendado: Para solucionar problemas menores sin interrumpir el funcionamiento de la instancia.

(What Is The Difference Between Terminating And Stopping An EC2 Instance?, s. f.)

¿Qué papel juega una AMI (Amazon Machine Image) al lanzar una instancia?

Una Amazon Machine Image (AMI) es un componente esencial para lanzar instancias en Amazon EC2, ya que proporciona el entorno de software necesario para iniciar y configurar una máquina virtual en la nube.

Al lanzar una instancia EC2, es obligatorio especificar una AMI, ya que esta contiene:

- El sistema operativo.
- Las aplicaciones preinstaladas.
- Las configuraciones de arranque necesarias.
- Una asignación de dispositivos de bloque, que determina los volúmenes que se adjuntarán a la instancia.

Una AMI está definida por varios parámetros, que determinan su compatibilidad y comportamiento:

- **Región:** Las AMIs son específicas de cada región de AWS.
- **Sistema operativo:** Por ejemplo, Amazon Linux, Ubuntu, Windows Server, entre otros.
- **Arquitectura del procesador:** ARM, x86, etc.
- **Tipo de dispositivo raíz:** Puede ser EBS o almacenamiento de instancia.
- **Tipo de virtualización:** HVM o PV (Paravirtualización, aunque está en desuso).

Usos y ventajas de las AMIs:

- **Reutilización de configuraciones:** Se pueden lanzar múltiples instancias desde una sola AMI para replicar entornos consistentes.
- **Personalización:** Es posible crear una AMI personalizada a partir de una instancia EC2 ya configurada.
- **Portabilidad:** Las AMIs pueden copiarse a otras regiones para desplegar instancias en distintas ubicaciones geográficas.
- **Colaboración:** Una AMI puede ser compartida con otras cuentas de AWS.
- **Comercialización:** Las AMIs también pueden ser vendidas a través del AWS Marketplace.

Tipos de AMIs disponibles:

- **AMIs proporcionadas por AWS:** Imágenes oficiales y mantenidas por AWS.
- **AMIs públicas:** Compartidas por otros usuarios y disponibles de forma gratuita.
- **AMIs privadas:** Creadas y utilizadas de forma interna dentro de una organización.
- **AMIs de terceros:** Disponibles a través del AWS Marketplace, pueden ser gratuitas o de pago.

(Amazon Machine Images In Amazon EC2 - Amazon Elastic Compute Cloud, s. f.)

¿En qué casos sería recomendable elegir una AMI diferente a la AMI predeterminada de Amazon Linux?

Aunque la AMI predeterminada de Amazon Linux es una opción comúnmente utilizada por su compatibilidad y optimización para servicios de AWS, existen múltiples escenarios en los

que es aconsejable seleccionar una AMI diferente, dependiendo de los requisitos específicos del entorno que se desea implementar.

Uno de los factores más importantes a considerar al elegir una AMI distinta es el sistema operativo. Si el proyecto requiere ejecutar aplicaciones o servicios que sólo son compatibles con Windows Server, Ubuntu, Red Hat Enterprise Linux, u otro sistema operativo específico, será necesario elegir una AMI que lo incluya desde el inicio.

Asimismo, la arquitectura del procesador puede influir en esta decisión. Por ejemplo, si se desea lanzar instancias optimizadas para arquitecturas ARM de 64 bits (como las instancias Graviton), se debe elegir una AMI compatible con esa arquitectura. De igual forma, si se utilizan aplicaciones legadas que sólo funcionan en arquitecturas de 32 bits, se requerirá una AMI que lo soporte.

También es importante considerar el tipo de dispositivo raíz que utiliza la AMI. Algunas AMIs están respaldadas por Amazon EBS, lo que permite detener e iniciar las instancias sin perder datos. Otras, en cambio, utilizan almacenamiento efímero (instance store), que borra todos los datos al detener o finalizar la instancia. La elección dependerá de las necesidades de persistencia y recuperación del sistema.

Otro caso en el que conviene optar por una AMI diferente es cuando se requiere software adicional preinstalado, como bases de datos (por ejemplo, SQL Server), entornos de desarrollo o herramientas específicas de terceros. En estos casos, se pueden seleccionar AMIs preconfiguradas disponibles en el AWS Marketplace o proporcionadas por otros usuarios.

Además, las AMIs son específicas de cada región de AWS, por lo que si se desea lanzar instancias en una región diferente, se deberá verificar la disponibilidad de la AMI correspondiente o copiar una AMI personalizada a esa región.

(Find An AMI That Meets The Requirements For Your EC2 Instance - Amazon Elastic Compute Cloud, s. f.)

¿Cómo agregar y adjuntar volúmenes, distinguir entre EBS y almacenamiento efímero?

Amazon EC2 permite adjuntar volúmenes de almacenamiento adicionales a las instancias para ampliar su capacidad. El tipo más común de almacenamiento es Amazon EBS (Elastic Block Store), que proporciona almacenamiento persistente, a diferencia del almacenamiento efímero, que se pierde al detener o terminar una instancia.

Los volúmenes EBS deben estar en la misma zona de disponibilidad (Availability Zone) que la instancia a la que se desea adjuntar. Es posible adjuntar varios volúmenes a una misma instancia, dependiendo del tipo de instancia utilizado, ya que cada tipo tiene un límite máximo de volúmenes adjuntos. Si se supera este límite, la solicitud de adjuntar el volumen fallará con un error del tipo AttachmentLimitExceeded. En situaciones específicas, los volúmenes EBS con la funcionalidad Multi-Attach habilitada pueden ser conectados simultáneamente hasta a 16 instancias EC2, lo cual es útil para escenarios de alta disponibilidad o aplicaciones compartidas.

Además, existen ciertas restricciones cuando un volumen contiene un código de producto del AWS Marketplace. En estos casos, solo se puede adjuntar a una instancia que esté detenida, el usuario debe estar suscrito al producto asociado y la instancia debe ser compatible con el sistema operativo y configuración del volumen, por ejemplo, no se puede adjuntar un volumen de Windows a una instancia con Linux. En cuanto al proceso de adjuntar un volumen, puede hacerse desde la consola de EC2, mediante la AWS CLI (`attach-volume`) o usando PowerShell (`Add-EC2Volume`), especificando el ID del volumen, el ID de la instancia y el nombre del dispositivo (por ejemplo, `/dev/sdf`). Cabe destacar que el nombre del dispositivo usado por el sistema operativo puede diferir del especificado en la solicitud de adjunto, debido a cómo el sistema maneja los controladores de dispositivos.

Por último, es importante comprender que si se adjunta un volumen que fue el volumen raíz de otra instancia (por ejemplo, mediante un snapshot), este podría reemplazar el volumen raíz actual, lo que podría alterar el comportamiento del arranque de la instancia. En estos casos, se debe revisar cuidadosamente la configuración para evitar iniciar desde un volumen incorrecto. (*Attach An Amazon EBS Volume To An Amazon EC2 Instance - Amazon EBS*, s. f.)

¿Cómo ejecutar comandos remotos sin SSH?

Amazon EC2 Run Command, parte de AWS Systems Manager, permite ejecutar comandos remotos sobre instancias EC2 sin necesidad de utilizar SSH, lo cual mejora la seguridad, la trazabilidad y la eficiencia operativa. Esta herramienta está diseñada para administrar entornos de nube a gran escala, permitiendo ejecutar instrucciones en una o varias

instancias de forma centralizada, segura y auditable, sin requerir conexiones manuales, claves SSH ni configuraciones adicionales como saltos (jump boxes) o direcciones IP habilitadas.

Run Command se integra con IAM para controlar el acceso, lo que elimina la necesidad de gestionar claves SSH. Cada operación realizada se registra automáticamente en AWS CloudTrail, ofreciendo un historial detallado con el usuario que la ejecutó, los parámetros utilizados, el estado de ejecución y las instancias involucradas. Esta trazabilidad hace que Run Command sea más seguro y controlado en comparación con conexiones SSH tradicionales.

Además, Run Command permite gestionar múltiples instancias en paralelo, lo que resulta especialmente útil para tareas repetitivas o masivas, como reiniciar servicios, obtener archivos de logs o verificar el estado de componentes distribuidos. Las tareas complejas pueden ser estandarizadas a través de documentos personalizados, que encapsulan comandos o scripts reutilizables y gestionan el acceso de forma estructurada. Estos documentos pueden combinarse con AWS Lambda para crear flujos de automatización más avanzados.

(Manage Instances At Scale Without SSH Access Using EC2 Run Command | Amazon Web Services, 2022)

¿Qué pasos se requieren para adjuntar un volumen EBS adicional a una instancia de Linux existente?

Para adjuntar un volumen EBS adicional a una instancia EC2 con Linux, es necesario que el volumen se encuentre en el mismo Availability Zone que la instancia. Una vez creado el volumen desde la consola de Amazon EC2, CLI o API, se puede adjuntar a una instancia existente especificando el ID de volumen, el ID de instancia y el nombre del dispositivo (por

ejemplo, /dev/sdf). Este nombre puede variar a nivel del sistema operativo dependiendo del controlador de dispositivos utilizado.

Es importante tener en cuenta que el número máximo de volúmenes que se pueden adjuntar depende del tipo de instancia. Si se supera este límite, la solicitud fallará con un error `AttachmentLimitExceeded`. Además, si el volumen contiene un código de producto del AWS Marketplace, solo podrá ser adjuntado a instancias detenidas, y la instancia debe ser compatible con el sistema operativo del volumen. En entornos avanzados, los volúmenes EBS que tienen habilitada la opción Multi-Attach pueden ser conectados a hasta 16 instancias EC2 al mismo tiempo, lo cual resulta útil para aplicaciones distribuidas con acceso concurrente a datos.

Tras adjuntar el volumen, será necesario inicializarlo, formatearlo y montarlo manualmente en la instancia Linux, para que esté disponible como un sistema de archivos. Este proceso incluye la creación de un sistema de archivos (por ejemplo, con `mkfs`), la creación de un punto de montaje y el uso del comando `mount` para asociar el volumen al sistema. También se recomienda verificar que no se adjunte accidentalmente un volumen que funcione como disco raíz de otra instancia, ya que esto puede provocar conflictos durante el arranque si se intenta iniciar desde un volumen incorrecto. La configuración adecuada garantiza un proceso seguro, eficiente y alineado con las mejores prácticas de administración de almacenamiento en Amazon EC2. (*Attach An Amazon EBS Volume To An Amazon EC2 Instance - Amazon EBS*, s. f.)

¿Qué sucede con los datos de un volumen EBS cuando se detiene o finaliza la instancia?

adjuntos, especialmente los volúmenes EBS, depende de la configuración del atributo `DeleteOnTermination`. En el caso de volúmenes de almacenamiento efímero (instance store), los datos se eliminan automáticamente al terminar la instancia, ya que este tipo de almacenamiento no persiste más allá del ciclo de vida de la instancia. Por ello, si se desea conservar estos datos, se deben copiar manualmente a un almacenamiento persistente, como un volumen EBS, un bucket de Amazon S3 o un sistema de archivos Amazon EFS.

Para volúmenes Amazon EBS, el comportamiento varía según la configuración de eliminación al terminar. Si el atributo `DeleteOnTermination` está establecido en `"true"`, el volumen se elimina automáticamente al terminar la instancia. Por el contrario, si está en `"false"`, el volumen se conserva, y puede reutilizarse o respaldarse mediante snapshots, aunque continuará generando costos hasta que se elimine manualmente. Este atributo puede definirse tanto al lanzar la instancia como después, y también puede modificarse a través de la consola, CLI o PowerShell.

De manera predeterminada, los volúmenes raíz creados durante el lanzamiento de una instancia se configuran para eliminarse al terminar la misma. Sin embargo, los volúmenes raíz adjuntados después del lanzamiento se conservan por defecto. Para los volúmenes de datos (no raíz), si se adjuntan mediante la consola, tienden a preservarse, mientras que, si se usan comandos CLI, se configuran generalmente para eliminarse al terminar.

Es recomendable verificar y, si es necesario, modificar la configuración del atributo `DeleteOnTermination` al momento de lanzar una instancia o durante su ciclo de vida, para asegurar la persistencia de datos crítica después de apagar o terminar instancias EC2.

(Preserve Data When An Instance Is Terminated - Amazon Elastic Compute Cloud, s. f.)

¿Funcionará el shell que configuramos en Slackware para Linux en la instancia que creamos? ¿Qué necesito cambiar?

El shell que has configurado en tu entorno Linux Slackware puede no funcionar directamente en una instancia de Amazon Linux sin realizar algunos ajustes previos. Hay varias diferencias clave entre ambas plataformas que debes tener en cuenta.

Primero, Slackware no está disponible como una AMI predeterminada en AWS; para utilizar Slackware en EC2, sería necesario crear una imagen personalizada desde cero, lo cual implica preparar el sistema (como un `.vmdk`), ajustarlo para arrancar en EC2 y configurar el arranque, el `fstab` con `UUID`, entre otros. Esto sugiere que el entorno Slackware no es estándar en AWS y requiere esfuerzo considerable para migrar.

En comparación, Amazon Linux es una distribución distinta, basada en RPM, optimizada para AWS, soportada oficialmente y mantenida con parches de seguridad por AWS. Si tu configuración de shell en Slackware depende del uso de herramientas o utilidades específicas, como `chsh`, es posible que debas instalarlas manualmente en Amazon Linux. Por ejemplo, para cambiar el shell en Amazon Linux 2 (por ejemplo a `zsh`), primero hay que instalar el paquete `linux-user` antes de usar `chsh`.

Además, Amazon Linux está diseñado para integrarse con servicios AWS y puede incluir dependencias o comportamientos diferentes a los de Slackware, especialmente en componentes de sistema como init, paths, kernels y sistema de paquetes.

El shell que configuramos en Slackware es compatible con otras distribuciones de Linux como Amazon Linux, que es el sistema operativo de la instancia EC2 que creamos en AWS. Esto se debe a que ambos sistemas utilizan Bash como intérprete de comandos, por lo que los scripts escritos en este lenguaje deberían funcionar sin problemas. Sin embargo, es importante verificar que los comandos utilizados en el script estén disponibles en Amazon Linux, ya que algunas herramientas específicas de Slackware podrían no estar preinstaladas. En ese caso, sería necesario instalar las dependencias faltantes usando el gestor de paquetes correspondiente. También se deben revisar los permisos de ejecución del script y asegurarse de que las rutas o configuraciones utilizadas sean compatibles con el entorno de la instancia. Para comprobar su funcionalidad, se subió el script a la instancia EC2 mediante el comando scp desde PowerShell en Windows, y luego se ejecutó dentro de la instancia usando SSH. Esta prueba permite validar que el script funciona correctamente en un entorno diferente al original, demostrando su portabilidad y utilidad en sistemas basados en Linux.

Conclusiones

A lo largo del desarrollo del laboratorio, se abordaron de forma práctica y progresiva diversos aspectos fundamentales en la administración de sistemas operativos y redes, consolidando conocimientos teóricos previamente adquiridos. Se logró completar con éxito la instalación y configuración básica de sistemas operativos en máquinas físicas, virtuales y en la nube, lo cual permitió comprender mejor las diferencias entre entornos locales y cloud computing.

Además, se trabajó con herramientas de simulación y análisis de red como Packet Tracer y Wireshark, lo que facilitó el entendimiento del funcionamiento de los protocolos de red, la encapsulación de datos en diferentes capas, y la importancia de los mensajes ICMP para la conectividad entre dispositivos. También se profundizó en la estructura de las tarjetas de red, comparando dispositivos reales con máquinas virtuales, destacando similitudes y diferencias en sus configuraciones.

En la parte de administración mediante Shell Scripting, se desarrollaron scripts que automatizan tareas comunes de gestión de archivos, usuarios y análisis de logs en sistemas Unix/Linux, lo cual representa un avance significativo hacia la administración eficiente y segura de entornos productivos. Adicionalmente, se reforzó el uso del editor VI, una herramienta clave en entornos Unix, practicando comandos esenciales para la edición y manipulación de archivos de texto.

Finalmente, se estableció la configuración de un servicio de compartición de archivos SMB/Samba entre diferentes sistemas operativos, demostrando la capacidad de interoperabilidad entre plataformas heterogéneas dentro de una infraestructura empresarial.

Este laboratorio no solo fortaleció nuestras habilidades técnicas, sino que también mejoró nuestro trabajo en equipo y nuestra capacidad para enfrentar escenarios reales del ámbito profesional en redes y administración de sistemas.

Referencias

3.3. *The Main window.* (s. f.).

https://www.wireshark.org/docs/wsug_html_chunked/ChUseMainWindowSection.html

6.3. *Filtering packets while viewing.* (s. f.).

https://www.wireshark.org/docs/wsug_html_chunked/ChWorkDisplayFilterSection.html

6.6. *Defining and saving filters.* (s. f.).

https://www.wireshark.org/docs/wsug_html_chunked/ChWorkDefineFilterSection.html

Amazon Machine Images in Amazon EC2 - Amazon Elastic Compute Cloud. (s. f.).

<https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AMIs.html>

Attach an Amazon EBS volume to an Amazon EC2 instance - Amazon EBS. (s. f.).

<https://docs.aws.amazon.com/ebs/latest/userguide/ebs-attaching-volume.html>

Awati, R. (2025, 6 marzo). *What is promiscuous mode in networking?* Search Security.

<https://www.techtarget.com/searchsecurity/definition/promiscuous-mode>

Chapter 1. Introduction. (s. f.).

https://www.wireshark.org/docs/wsug_html_chunked/ChapterIntroduction.html#ChIntro
[WhatIs](#)

Cisco Networking Academy. (n.d.). *Cisco Packet Tracer: A free and fun course for beginners.*

Cisco. <https://www.netacad.com/courses/getting-started-cisco-packet-tracer?courseLang=en-US>

features. (s. f.). Amazon Web Services, Inc.

<https://aws.amazon.com/ec2/features/?refid=f7bdb05e-2bc2-4bf3-99dd-ed8091461c9e#topic-0>

Find an AMI that meets the requirements for your EC2 instance - Amazon Elastic Compute Cloud. (s. f.). <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/finding-an-ami.html>

Mack, H. (2025, 30 agosto). *Straight-Through vs. Crossover Cables: What's the Difference?* Uprite IT Services. <https://www.uprite.com/straight-through-vs-crossover-cables/>

Manage Instances at Scale without SSH Access Using EC2 Run Command | Amazon Web Services. (2022, 3 noviembre). Amazon Web Services. <https://aws.amazon.com/blogs/aws/manage-instances-at-scale-without-ssh-access-using-ec2-run-command/>

Moolenaar, B. (n.d.). *VIM REFERENCE MANUAL.* Vimhelp. <https://vimhelp.org/motion.txt.html#word-motions>

Preserve data when an instance is terminated - Amazon Elastic Compute Cloud. (s. f.). <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/preserving-volumes-on-termination.html>

What is Amazon EC2? - Amazon Elastic Compute Cloud. (s. f.). <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/concepts.html>

What is the difference between terminating and stopping an EC2 instance? (s. f.).

https://docs.rightscale.com/faq/clouds/aws/Whats_the_difference_between_Terminating_and_Stopping_an_EC2_Instance.html

What is the OSI Model? (s. f.). Cloudflare.

<https://www.cloudflare.com/learning/ddos/glossary/open-systems-interconnection-model-osi/>