

INSTITUTO TÉCNICO SALESIANO VILLADA

LABORATORIO DE INFORMÁTICA

PROFESOR: *CALFIN*, Gustavo

CURSO: 7mo “C”

## **INFORME RASPBERRY - DHCP**

ALUMNOS:

*ASIS, Tomas*

*FAUS, Francisco*

*HLAVACH, Joaquin*

*GALLEGUILLO, Lucas*

*GIRARDI, Facundo*

*GUERRA, Matias*

*MENDEZ, Fabricio*

## Índice

<b>1. Introducción.....</b>	<b>2</b>
1.1. Justificación del Proyecto.....	2
<b>2. Raspberry Pi: Características y Utilización.....</b>	<b>3</b>
2.1. Descripción General de la Raspberry Pi.....	3
2.2. Hardware y Conectividad.....	4
2.3. Sistema Operativo Utilizado.....	5
<b>3. Términos Técnicos a Utilizar.....</b>	<b>6</b>
3.1. NAT (Network Address Translation).....	6
3.2. DHCP (Dynamic Host Configuration Protocol).....	6
3.3. DNS (Domain Name System).....	7
3.4. Gateway (Puerta de Enlace).....	7
3.5. IP Forwarding (Reenvío de IP).....	8
3.6. iptables.....	8
3.7. Switch de Red.....	8
<b>4. Especificaciones del Entorno de Trabajo.....</b>	<b>8</b>
4.1. Componentes Utilizados.....	9
4.1.1. Raspberry Pi.....	9
4.1.2. Switch de Red TP-Link TL-SF1008D.....	9
4.1.3. Notebooks.....	10
4.2. Software Utilizado.....	10
4.2.1. Sistema Operativo: Raspberry Pi OS (Raspbian).....	10
4.2.2. Servidor DHCP: ISC DHCP Server.....	10
4.2.3. iptables.....	10
<b>5. Resolución de las Consignas.....</b>	<b>11</b>
5.1. Configuración de las Interfaces de Red.....	11
5.2. Configuración del Servidor DHCP.....	12
5.3. Configuración del Reenvío de IPs (IP Forwarding).....	12
5.4. Configuración de iptables para NAT.....	13
5.5. Configuración de DNS.....	13
<b>6. Pruebas y Resultados.....</b>	<b>13</b>
<b>7. Imágenes Del Resultado.....</b>	<b>14</b>
<b>8. Defensa Oral.....</b>	<b>15</b>
8.1. Descripción de la Defensa Oral.....	16
8.2. Preguntas Realizadas por el Profesor.....	16
8.2.1. Pregunta 1: Medir transferencia de datos (Logrado).....	16
8.2.2. Pregunta 2: Bloquear una IP (Logrado).....	16
8.2.3. Pregunta 3: Revisar funcionamiento del DHCP (Logrado).....	16
8.2.4. Pregunta 4: Conexión a Internet a través del DHCP (Logrado).....	17
8.2.5. Pregunta 5: Bloquear una página web (Parcialmente Logrado).....	17
8.2.6. Pregunta 6: Limitar velocidad de transferencia de datos (No Logrado).....	17
8.2.7. Pregunta 7: Guardar reglas de iptables (Logrado).....	17
8.2.8. Pregunta 8: Crear script de inicio para iptables (No Logrado).....	17
8.2.9. Pregunta 9: Pasar de IPv4 a IPv6 (Parcialmente Logrado).....	18
8.2.10. Pregunta 10: Conectar Wi-Fi (No Logrado).....	19
<b>9. Conclusiones.....</b>	<b>19</b>
9.1. Evaluación General del Proyecto.....	20

## 1. Introducción

El presente proyecto se enfoca en la implementación y configuración de una Raspberry Pi como servidor DHCP (Dynamic Host Configuration Protocol) para redes basadas en IPv4. Este protocolo es esencial en redes modernas, ya que permite asignar automáticamente direcciones IP a dispositivos conectados a una red, facilitando la administración y garantizando que cada dispositivo reciba una dirección única y válida.

Utilizando una Raspberry Pi, un dispositivo de bajo costo y gran versatilidad, se busca implementar una solución eficiente para gestionar la asignación de direcciones IP y otros parámetros de red, como servidores DNS y la puerta de enlace predeterminada. Esta configuración es ideal para entornos de red pequeños o medianos, como laboratorios, oficinas o redes domésticas, donde se requiere automatización y gestión centralizada de la red.

### 1.1. Justificación del Proyecto

El uso de un servidor DHCP en redes locales es una práctica esencial que permite simplificar la gestión de las direcciones IP, evitando conflictos de IP y errores manuales en la configuración. Este proyecto está justificado por varias razones técnicas y prácticas:

**Automatización de la Asignación de IPs:** La configuración manual de direcciones IP en una red puede generar problemas de configuración, como la asignación incorrecta de IPs o conflictos entre dispositivos. Un servidor DHCP elimina estos problemas, garantizando que cada dispositivo conectado a la red reciba una dirección IP válida de forma automática, facilitando la administración de la red.

**Optimización del Tiempo y Recursos:** El uso de una Raspberry Pi como servidor DHCP permite implementar una solución eficiente y económica en comparación con hardware de red dedicado más costoso. Esto hace que sea una opción ideal para entornos educativos, redes domésticas o pequeñas empresas que requieren una gestión sencilla pero efectiva de su red.

**Escalabilidad y Flexibilidad:** Aunque el proyecto inicial está pensado para gestionar redes pequeñas o medianas, la solución puede escalar a redes más grandes ajustando el rango de IPs y añadiendo más funcionalidades al servidor DHCP en la Raspberry Pi.

**Seguridad y Control:** Al gestionar la asignación de direcciones IP desde un único servidor centralizado, se pueden implementar políticas de seguridad y control de acceso. Esto permite restringir el acceso de ciertos dispositivos, bloquear dominios específicos y monitorear la actividad de la red para garantizar su seguridad y correcto funcionamiento.

**Preparación para el Futuro:** Aunque este proyecto se enfoca en el protocolo IPv4, la misma infraestructura puede adaptarse para gestionar redes basadas en IPv6, lo que garantiza que la solución pueda evolucionar conforme crezcan las necesidades de la red y se adopten nuevos estándares.

## **2. Raspberry Pi: Características y Utilización**

La Raspberry Pi es un ordenador de bajo costo y pequeño tamaño que ha sido ampliamente adoptado en diversas áreas, como la educación, la investigación, proyectos de redes, robótica y automatización, gracias a su versatilidad y potencia. Con su capacidad de ejecutar sistemas operativos basados en Linux, la Raspberry Pi se ha convertido en una herramienta clave para desarrolladores, estudiantes y entusiastas de la tecnología, quienes la emplean en una variedad de proyectos que van desde servidores domésticos hasta sistemas de control industrial.

En el contexto de este proyecto, la Raspberry Pi es utilizada como un servidor DHCP (Protocolo de Configuración Dinámica de Host) para una red local. Esto permite automatizar la asignación de direcciones IPv4 a los dispositivos conectados a la red, facilitando la gestión y optimización del entorno de red.

### **2.1. Descripción General de la Raspberry Pi**

La Raspberry Pi es un ordenador de placa única (SBC, por sus siglas en inglés) desarrollado por la Fundación Raspberry Pi, con el objetivo inicial de promover la enseñanza de la informática en escuelas y países en desarrollo. Sin embargo, gracias a su funcionalidad y precio accesible, ha sido ampliamente adoptada en diferentes sectores para la creación de proyectos innovadores.

Existen varios modelos de Raspberry Pi, cada uno con diferentes especificaciones de hardware, pero todos mantienen un conjunto básico de características que incluyen una CPU basada en arquitectura ARM, memoria RAM, almacenamiento a través de tarjetas microSD y una amplia gama de puertos de conectividad.

La Raspberry Pi es ideal para utilizarse como un servidor o punto de control en pequeñas redes debido a su bajo consumo de energía, tamaño compacto y capacidad para ejecutar tareas que, en otros casos, requerirían hardware dedicado más costoso.

Entre los modelos más recientes se encuentran la Raspberry Pi 4 y la Raspberry Pi 400, que ofrecen una mayor capacidad de procesamiento y memoria, permitiendo su uso en tareas más exigentes, como la gestión de redes, servidores de medios o centros de control de IoT (Internet de las Cosas).

### **2.2. Hardware y Conectividad**

El hardware de la Raspberry Pi está diseñado para ser modular y flexible, lo que permite conectarle una variedad de dispositivos externos y periféricos, además de ofrecer la posibilidad de integrarse a redes tanto cableadas como inalámbricas.

Principales características de hardware:

Procesador: Dependiendo del modelo, las Raspberry Pi más recientes utilizan un procesador ARM Cortex-A72 de 64 bits y cuatro núcleos, lo que le permite ejecutar aplicaciones de alto rendimiento y tareas simultáneas.

Memoria RAM: Los modelos más recientes, como la Raspberry Pi 4, ofrecen hasta 8 GB de RAM, lo que facilita su uso en proyectos más exigentes y multitarea.

Almacenamiento: La Raspberry Pi utiliza tarjetas microSD como medio de almacenamiento principal, donde se instalan el sistema operativo y los archivos del usuario.

#### Puertos de entrada/salida:

USB: Modelos como la Raspberry Pi 4 incluyen puertos USB 3.0 y 2.0, que permiten la conexión de dispositivos como teclados, ratones, discos duros externos o adaptadores de red.

HDMI: La Pi cuenta con puertos micro HDMI para conectar monitores y pantallas.

Ethernet: Para la conectividad de red cableada, se incluyen puertos Gigabit Ethernet, lo que permite una conexión rápida y confiable a una red local o a Internet.

Wi-Fi y Bluetooth: Los modelos recientes incluyen conectividad Wi-Fi de doble banda y Bluetooth, lo que facilita la conexión inalámbrica a redes y dispositivos.

GPIO (General Purpose Input/Output): La Raspberry Pi cuenta con un conjunto de pines GPIO, que son usados para interactuar con otros dispositivos, como sensores, motores o LEDs, lo que amplía su capacidad para proyectos de electrónica y automatización.

Consumo de energía: Una de las ventajas más destacadas de la Raspberry Pi es su bajo consumo de energía, con un promedio de 5 a 10 watts, lo que la convierte en una opción ideal para proyectos que requieren estar en funcionamiento continuo sin generar altos costos energéticos.

#### Conectividad:

Redes Cableadas: A través del puerto Ethernet, la Raspberry Pi puede integrarse fácilmente en redes cableadas, lo que proporciona una conexión estable y rápida. Esto es fundamental en su rol como servidor DHCP.

Redes Inalámbricas: La conectividad Wi-Fi incorporada permite que la Raspberry Pi se conecte a redes inalámbricas, lo que puede ser útil en configuraciones donde el acceso a conexiones cableadas no es factible.

La combinación de estos elementos hace que la Raspberry Pi sea una plataforma perfecta para proyectos que requieren tanto capacidad de procesamiento como flexibilidad de conectividad, en especial para su uso como servidor en redes domésticas o de pequeña escala.

### **2.3. Sistema Operativo Utilizado**

El sistema operativo utilizado en la Raspberry Pi para este proyecto es Raspberry Pi OS (anteriormente conocido como Raspbian), una distribución basada en Debian Linux optimizada específicamente para el hardware de la Raspberry Pi.

Raspberry Pi OS es el sistema operativo recomendado oficialmente por la Fundación Raspberry Pi, y ofrece una amplia compatibilidad con el hardware de la placa, además de

una excelente estabilidad y seguridad. Se puede ejecutar en modo gráfico o en modo de línea de comandos, dependiendo de las necesidades del proyecto.

#### Características principales de Raspberry Pi OS:

Ligero y optimizado: A pesar de ser un sistema operativo completo basado en Linux, Raspberry Pi OS está optimizado para funcionar eficientemente en el hardware limitado de la Raspberry Pi, permitiendo un rendimiento fluido incluso en proyectos exigentes.

Compatibilidad con paquetes Debian: Como está basado en Debian, Raspberry Pi OS es compatible con la mayoría de los paquetes y aplicaciones de esta distribución, lo que permite instalar y ejecutar herramientas comunes de red y desarrollo, como el servidor DHCP utilizado en este proyecto.

Fácil instalación: Se instala de forma sencilla a través de una tarjeta microSD utilizando herramientas como Raspberry Pi Imager. Además, permite una configuración inicial rápida, lo que lo hace ideal para proyectos que requieren una implementación eficiente.

Acceso remoto: Al estar basado en Linux, Raspberry Pi OS facilita la administración remota de la Raspberry Pi a través de protocolos como SSH, lo que permite gestionar la configuración del servidor DHCP y otros servicios desde cualquier otro dispositivo conectado a la red.

En este proyecto, Raspberry Pi OS ha sido la base del entorno de trabajo para la configuración del servidor DHCP, proporcionando las herramientas necesarias para la gestión de la red y la automatización de la asignación de direcciones IP en el entorno local.

### **3. Términos Técnicos a Utilizar**

En la configuración de una Raspberry Pi como servidor de red y DHCP, es fundamental conocer los términos técnicos que definen el funcionamiento y los componentes clave de la red. A continuación, se describen algunos de los conceptos más importantes que se emplean en este proyecto:

#### **3.1. NAT (Network Address Translation)**

NAT (Network Address Translation) es una técnica utilizada en redes para modificar las direcciones IP en los paquetes de datos que atraviesan un router o dispositivo de red. Esencialmente, NAT permite que múltiples dispositivos en una red local (privada) utilicen una sola dirección IP pública para acceder a Internet.

Función de NAT en este proyecto: En este proyecto, la Raspberry Pi puede actuar como un punto de conexión que utiliza NAT para permitir que las notebooks conectadas a la red local comparten una sola dirección IP pública para acceder a recursos en Internet.

### Ventajas de NAT:

Conservación de direcciones IP públicas: Dado que las direcciones IPv4 públicas son limitadas, NAT permite que varios dispositivos usen una sola IP pública para la comunicación externa.

Seguridad básica: NAT oculta las direcciones IP internas de la red local, proporcionando una capa básica de seguridad al evitar que las direcciones IP internas sean visibles desde Internet.

NAT se implementa mediante herramientas como iptables en Linux, y en este proyecto se configura en la Raspberry Pi para permitir que todos los dispositivos conectados accedan a Internet a través de la misma IP pública.

### **3.2. DHCP (Dynamic Host Configuration Protocol)**

DHCP (Dynamic Host Configuration Protocol) es un protocolo de red utilizado para asignar dinámicamente direcciones IP y otros parámetros de configuración de red a dispositivos en una red. Un servidor DHCP, como la Raspberry Pi en este caso, automatiza el proceso de asignación de direcciones IP, evitando la necesidad de configurar manualmente cada dispositivo.

Función de DHCP en este proyecto: El servidor DHCP en la Raspberry Pi asigna automáticamente direcciones IP a las notebooks conectadas al switch. Al hacerlo, también puede proporcionar la dirección de la puerta de enlace, los servidores DNS, y otra información importante para que los dispositivos se integren en la red.

### Ventajas del DHCP:

Automatización: Evita la necesidad de configurar manualmente direcciones IP en cada dispositivo.

Evitar conflictos de IP: El servidor DHCP asegura que no haya dos dispositivos con la misma dirección IP en la red.

### **3.3. DNS (Domain Name System)**

DNS (Domain Name System) es el sistema que traduce nombres de dominio legibles por humanos (como google.com) en direcciones IP que las máquinas puedan utilizar para comunicarse. Esto es fundamental, ya que los dispositivos en una red no pueden usar nombres de dominio directamente, sino que requieren una dirección IP para acceder a servidores y sitios web.

Función del DNS en este proyecto: El servidor DHCP en la Raspberry Pi puede proporcionar la dirección del servidor DNS a los dispositivos conectados para que puedan resolver nombres de dominio. Generalmente, se utilizan servidores DNS públicos como los de Google (8.8.8.8) o Cloudflare (1.1.1.1).

### Ventajas del DNS:

Simplificación: Permite a los usuarios acceder a sitios web por sus nombres en lugar de tener que recordar direcciones IP numéricas.

Flexibilidad: Permite que las direcciones IP cambien sin afectar a los usuarios, ya que siempre acceden a través del nombre de dominio.

### **3.4. Gateway (Puerta de Enlace)**

La puerta de enlace es un dispositivo de red que actúa como el punto de acceso entre una red interna y otra red, generalmente una red externa como Internet. En una red local, los dispositivos necesitan conocer la dirección IP de la puerta de enlace para poder enviar tráfico hacia fuera de la red local, es decir, hacia Internet.

Función de la puerta de enlace en este proyecto: En este caso, la Raspberry Pi actúa como la puerta de enlace para las notebooks conectadas al switch. Esto significa que todo el tráfico de red de las notebooks con destino a Internet pasa primero por la Raspberry Pi antes de salir de la red local.

#### Importancia de la puerta de enlace:

Si una puerta de enlace configurada correctamente, los dispositivos en una red local no pueden comunicarse con dispositivos fuera de esa red.

### **3.5. IP Forwarding (Reenvío de IP)**

El IP Forwarding (Reenvío de IP) es una función del sistema operativo que permite que un dispositivo con múltiples interfaces de red (como una Raspberry Pi) reenvíe paquetes de datos de una interfaz a otra. Es esencial para que un dispositivo actúe como un router entre diferentes redes.

Función del reenvío de IP en este proyecto: En este proyecto, el reenvío de IP se habilita en la Raspberry Pi para que pueda transferir tráfico entre la red local (a través de eth0) y la red externa (a través de eth1 o una conexión a Internet). Esto es necesario para que las notebooks conectadas puedan acceder a Internet.

### **3.6. iptables**

iptables es una herramienta de firewall y filtrado de paquetes para sistemas basados en Linux. Se utiliza para controlar el tráfico de red que entra y sale de una máquina, permitiendo crear reglas que filtren, dirijan o bloquen paquetes de datos según distintos criterios.

Función de iptables en este proyecto: En el proyecto, iptables se utiliza para implementar NAT, reenvío de paquetes, y restricciones de tráfico. Es una herramienta clave para asegurar que el tráfico de la red local pueda salir a Internet y regresar a los dispositivos correctos.

Reglas personalizadas: Con iptables también se pueden bloquear ciertos dominios, restringir el acceso a subredes o limitar el tráfico según la cantidad de datos transmitidos.

### **3.7. Switch de Red**

Un switch de red es un dispositivo que permite conectar múltiples dispositivos en una red local. El switch actúa como un concentrador que distribuye el tráfico entre los dispositivos conectados, permitiendo que se comuniquen entre sí.

Cómo funciona el switch en este proyecto: En este proyecto, el switch TP-Link TL-SF1008D conecta todas las notebooks a la Raspberry Pi. Las notebooks se comunican con la Raspberry Pi a través del switch, y la Raspberry Pi les asigna direcciones IP y les proporciona acceso a Internet. El switch no gestiona ni asigna IPs, simplemente distribuye el tráfico entre los dispositivos conectados.

## **4. Especificaciones del Entorno de Trabajo**

Para la realización de este proyecto, se han empleado tanto componentes de hardware como software específicos, diseñados para cumplir con los requisitos de configuración de una red local gestionada por una Raspberry Pi que actúa como servidor DHCP y NAT. A continuación, se detallan los componentes y software utilizados para la implementación del proyecto.

### **4.1. Componentes Utilizados**

El proyecto requiere una serie de dispositivos que permitirán la creación de una red local funcional, donde la Raspberry Pi gestiona la asignación de direcciones IP y el acceso a Internet de las notebooks conectadas al sistema.

#### **4.1.1. Raspberry Pi**

La Raspberry Pi es el dispositivo central de este proyecto. Actúa como el servidor principal que gestionará tanto el servidor DHCP para la asignación de direcciones IP dinámicas como el NAT (Network Address Translation), lo que permite que los dispositivos conectados a la red local comparten una única conexión a Internet.

##### Funciones en el Proyecto:

Servidor DHCP: La Raspberry Pi asignará automáticamente direcciones IP a los dispositivos conectados mediante el servidor DHCP.

Puerta de enlace (gateway): La Raspberry Pi funcionará como la puerta de enlace predeterminada, gestionando el tráfico que se envía hacia Internet.

NAT: Implementará NAT utilizando iptables para permitir que las notebooks accedan a Internet a través de una única dirección IP pública.

#### **4.1.2. Switch de Red TP-Link TL-SF1008D**

El switch de red TP-Link TL-SF1008D es un dispositivo de red no gestionado con 8 puertos, que permite la conexión simultánea de varios dispositivos. En este proyecto, es el punto de conexión central para las notebooks y la Raspberry Pi.

Modelo: TP-Link TL-SF1008D.

#### Funciones en el Proyecto:

El switch permite que múltiples notebooks se conecten a la red y reciba direcciones IP desde la Raspberry Pi.

Facilita la distribución del tráfico de red hacia la Raspberry Pi y el intercambio de datos entre los dispositivos conectados.

Capacidad: Soporta hasta 8 dispositivos conectados simultáneamente mediante cables Ethernet, con una velocidad de transferencia de hasta 100 Mbps en cada puerto.

#### **4.1.3. Notebooks**

Varias notebooks son utilizadas como clientes en la red, conectándose al switch de red y recibiendo direcciones IP asignadas dinámicamente por el servidor DHCP en la Raspberry Pi.

#### Funciones en el Proyecto:

Cada notebook recibirá una dirección IP única y será capaz de acceder a Internet a través de la conexión gestionada por la Raspberry Pi.

Las notebooks servirán para realizar pruebas de conectividad y verificar el correcto funcionamiento del servidor DHCP y del NAT.

### **4.2. Software Utilizado**

El software utilizado en este proyecto está basado en herramientas de código abierto disponibles para el sistema operativo Linux, lo que permite la gestión y configuración completa de la red desde la Raspberry Pi.

#### **4.2.1. Sistema Operativo: Raspberry Pi OS (Raspbian)**

Raspberry Pi OS (anteriormente conocido como Raspbian) es una distribución de Linux basada en Debian, especialmente optimizada para el hardware de la Raspberry Pi. Ofrece una interfaz simple y acceso a todas las herramientas necesarias para la gestión de redes y servidores.

#### Funciones en el Proyecto:

Proporciona el entorno de trabajo base para configurar el servidor DHCP y el NAT.

Soporta la instalación de herramientas como isc-dhcp-server e iptables para la configuración de redes.

#### **4.2.2. Servidor DHCP: ISC DHCP Server**

El ISC DHCP Server es un servidor de DHCP de código abierto utilizado para asignar direcciones IP de forma dinámica a los dispositivos que se conectan a la red. En este proyecto, es instalado y configurado en la Raspberry Pi.

#### Funciones en el Proyecto:

Gestiona la asignación de direcciones IP, puerta de enlace predeterminada y servidores DNS a los dispositivos conectados a la red.

Automatiza el proceso de configuración de red de las notebooks, evitando configuraciones manuales y posibles conflictos de IP.

#### **4.2.3. iptables**

iptables es una herramienta de firewall y filtrado de paquetes para sistemas Linux, utilizada en este proyecto para implementar NAT (Network Address Translation) y controlar el tráfico de red entre las interfaces de la Raspberry Pi.

##### Funciones en el Proyecto:

Implementa NAT para que todas las notebooks puedan acceder a Internet a través de la conexión gestionada por la Raspberry Pi.

Permite aplicar reglas de filtrado y control de tráfico, como el bloqueo de dominios o el tráfico de determinados dispositivos.

### **5. Resolución de las Consignas**

A continuación, se detallan los pasos tomados para resolver los problemas identificados durante la configuración de la Raspberry Pi como servidor DHCP y la gestión de la conexión a Internet para los dispositivos conectados.

#### **5.1. Configuración de las Interfaces de Red**

Para que la Raspberry Pi gestione tanto la red local como la conexión a Internet, fue necesario configurar las interfaces de red. La Raspberry Pi cuenta con dos interfaces de red: una conectada a la red local (donde están las notebooks) y otra conectada a la red externa para obtener acceso a Internet.

Se configuraron las interfaces de red eth0 (para la red local) y eth1 (para la red externa) en el archivo **/etc/network/interfaces** de la Raspberry Pi de la siguiente manera:

```
source /etc/network/interfaces.d/*  
  
auto lo  
iface lo inet loopback  
  
# Configuración de eth0 para la red local (hacia la notebook)  
auto eth0  
iface eth0 inet static  
address 192.168.1.1  
netmask 255.255.255.0  
  
# Configuración de eth1 para obtener Internet de la red externa  
auto eth1  
iface eth1 inet dhcp
```

eth0 tiene una dirección IP estática en la red 192.168.1.0/24, lo que asegura que las notebooks se puedan comunicar con la Raspberry Pi.

eth1 está configurada para obtener una dirección IP dinámica (vía DHCP) de la red externa (en este caso, 192.168.30.0/24), lo que permite que la Raspberry Pi tenga acceso a Internet.

## 5.2. Configuración del Servidor DHCP

Para automatizar la asignación de direcciones IP a los dispositivos conectados a la red local, se instaló y configuró el servidor ISC DHCP en la Raspberry Pi. Este servidor se encarga de asignar direcciones IP dentro de un rango especificado, junto con otros parámetros de red como la puerta de enlace y los servidores DNS.

El archivo de configuración **/etc/dhcp/dhcpd.conf** fue modificado para incluir lo siguiente:

```
subnet 192.168.1.0 netmask 255.255.255.0 {  
    range 192.168.1.2 192.168.1.50;  
    option domain-name-servers 8.8.8.8, 1.1.1.1;  
    option routers 192.168.1.1;  
}
```

Esta configuración define:

Rango de IPs: Las direcciones IP serán asignadas dinámicamente en el rango 192.168.1.2 a 192.168.1.50.

Servidores DNS: Se utilizan los servidores DNS públicos de Google (8.8.8.8) y Cloudflare (1.1.1.1).

Puerta de enlace: La dirección de la Raspberry Pi (192.168.1.1) se establece como la puerta de enlace para las notebooks.

## 5.3. Configuración del Reenvío de IPs (IP Forwarding)

Para permitir que las notebooks conectadas a la red local puedan acceder a Internet a través de la Raspberry Pi, se habilitó el reenvío de IPs (IP Forwarding). Este paso es fundamental para que la Raspberry Pi actúe como un router entre la red local y la red externa.

El reenvío de IPs se habilitó temporalmente con el siguiente comando:

```
sudo sysctl -w net.ipv4.ip_forward=1
```

Para hacer que el reenvío de IPs sea permanente, se añadió la siguiente línea al archivo **/etc/sysctl.conf**.

```
net.ipv4.ip_forward=1
```

Esto asegura que el reenvío de IPs esté habilitado cada vez que se reinicie la Raspberry Pi.

#### 5.4. Configuración de iptables para NAT

Para que las notebooks accedan a Internet a través de la Raspberry Pi, se configuró NAT (Network Address Translation) utilizando iptables. NAT permite que varias notebooks en la red local comparten la misma conexión a Internet utilizando la IP pública de la Raspberry Pi.

Se ejecutaron los siguientes comandos para configurar NAT y permitir el tráfico entre las interfaces eth0 (red local) y eth1 (red externa):

```
sudo iptables -t nat -A POSTROUTING -o eth1 -j MASQUERADE  
sudo iptables -A FORWARD -i eth0 -o eth1 -j ACCEPT  
sudo iptables -A FORWARD -i eth1 -o eth0 -m state --state  
RELATED,ESTABLISHED -j ACCEPT
```

MASQUERADE: Permite que todas las conexiones salientes de la red local hacia la red externa utilicen la misma dirección IP pública.

FORWARD rules: Permiten el reenvío de tráfico entre las dos interfaces de red.

#### 5.5. Configuración de DNS

El archivo `/etc/resolv.conf` fue configurado para utilizar servidores DNS públicos, lo que asegura que tanto la Raspberry Pi como las notebooks puedan resolver correctamente los nombres de dominio. La configuración es la siguiente:

```
nameserver 8.8.8.8  
nameserver 1.1.1.1
```

Esto asegura que las solicitudes DNS de las notebooks y de la Raspberry Pi se resuelvan a través de los servidores DNS de Google y Cloudflare.

### 6. Pruebas y Resultados

Después de completar la configuración, se realizaron diversas pruebas para verificar que tanto la asignación de direcciones IP como el acceso a Internet funcionaban correctamente.

Ping a la puerta de enlace de la red externa:

Se realizó una prueba de conectividad hacia la red externa para verificar que la Raspberry Pi tenía acceso a Internet:

```
ping 192.168.30.1
```

El resultado confirmó que la Raspberry Pi podía comunicarse con la red externa.

Ping a un servidor DNS público (Google):

Se verificó que la Raspberry Pi tenía acceso a Internet al realizar un ping al servidor DNS público de Google:

```
ping 8.8.8.8
```

El resultado fue exitoso, lo que confirma que la conexión a Internet estaba activa.

#### Prueba de resolución DNS:

Se realizó una prueba de resolución DNS para verificar que los nombres de dominio se resolvían correctamente:

```
ping google.com
```

El resultado fue exitoso, confirmando que la Raspberry Pi y las notebooks podían resolver nombres de dominio.

#### Conexión desde una notebook:

Se conectó una notebook a la red local y se verificó que recibía una dirección IP asignada por el servidor DHCP. Además, se probó la conectividad a Internet realizando ping tanto a la Raspberry Pi (192.168.1.1) como a servidores externos (8.8.8.8). Todos los resultados fueron satisfactorios.

## 7. Imágenes Del Resultado

The terminal window shows the following output:

```
facu@raspberrypi:~$ sudo systemctl status isc-dhcp-server
● isc-dhcp-server.service - LSB: DHCP server
  Loaded: loaded (/etc/init.d/isc-dhcp-server; generated)
  Active: active (running) since Mon 2024-09-09 18:29:46 BST; 35min ago
    Docs: man:systemd-sysv-generator(8)
  Process: 744 ExecStart=/etc/init.d/isc-dhcp-server start (code=exited, status=0/SUCCESS)
    Tasks: 1 (limit: 762)
      CPU: 245ms
     CGroup: /system.slice/isc-dhcp-server.service
             └─757 /usr/sbin/dhcpcd -4 -q -cf /etc/dhcp/dhcpcd.conf eth0

Sep 09 18:47:29 raspberrypi dhcpcd[757]: DHCPOffer on 192.168.1.20 to f8:0d:ac:4c:92:fd (octalucero-HP-240-G7-Notebook-PC) via eth0
Sep 09 18:50:42 raspberrypi dhcpcd[757]: DHCPREQUEST for 192.168.1.3 from d0:37:45:be:33:c3 (facu-notebook) via eth0
Sep 09 18:50:42 raspberrypi dhcpcd[757]: DHCPPACK on 192.168.1.3 to d0:37:45:be:33:c3 (facu-notebook) via eth0
Sep 09 18:53:10 raspberrypi dhcpcd[757]: reuse_lease: lease age 148 (secs) under 25x threshold, reply with unaltered, existing lease for 192.168.1.3
Sep 09 18:53:10 raspberrypi dhcpcd[757]: DHCPREQUEST for 192.168.1.3 from d0:37:45:be:33:c3 (facu-notebook) via eth0
Sep 09 18:53:10 raspberrypi dhcpcd[757]: DHCPPACK on 192.168.1.3 to d0:37:45:be:33:c3 (facu-notebook) via eth0
Sep 09 18:53:52 raspberrypi dhcpcd[757]: DHCPREQUEST for 192.168.1.3 from d0:37:45:be:33:c3 (facu-notebook) via eth0
Sep 09 18:53:52 raspberrypi dhcpcd[757]: DHCPPACK on 192.168.1.3 from d0:37:45:be:33:c3 (facu-notebook) via eth0
Sep 09 18:53:52 raspberrypi dhcpcd[757]: DHCPREQUEST for 192.168.1.3 from d0:37:45:be:33:c3 (facu-notebook) via eth0
Sep 09 18:53:52 raspberrypi dhcpcd[757]: DHCPPACK on 192.168.1.3 from d0:37:45:be:33:c3 (facu-notebook) via eth0
Sep 09 19:00:36 raspberrypi dhcpcd[757]: DHCPREQUEST for 192.168.1.3 from d0:37:45:be:33:c3 (facu-notebook) via eth0
Sep 09 19:00:36 raspberrypi dhcpcd[757]: DHCPPACK on 192.168.1.3 to d0:37:45:be:33:c3 (facu-notebook) via eth0
facu@raspberrypi:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
4 bytes from 8.8.8.8: icmp_seq=1 ttl=115 time=12.6 ns
4 bytes from 8.8.8.8: icmp_seq=2 ttl=115 time=12.3 ns
```

La imagen muestra la terminal de una Raspberry Pi, donde se está ejecutando el servicio **isc-dhcp-server** para asignar direcciones IP mediante DHCP. En la parte superior de la captura se puede ver el estado del servicio, que está **activo** y corriendo correctamente. Además, se observa en el recuadro rojo, como se le asignó correctamente una ip a la notebook de prueba y además que tiene internet la raspberry.

The terminal window displays two sections of command-line output. The top section shows network interface statistics for 'enxd03745be33c3' with an MTU of 1500, IP 192.168.1.3, and various packet counts. The bottom section shows a ping test to 8.8.8.8 with round-trip times ranging from 12.8 ms to 13.2 ms.

```
enxd03745be33c3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 192.168.1.3 netmask 255.255.255.0 broadcast 192.168.1.255
inet6 fe80::bc4b:6120:54f1:932e prefixlen 64 scopeid 0x20<link>
ether d0:37:45:be:33:c3 txqueuelen 1000 (Ethernet)
RX packets 113913 bytes 123884645 (123.8 MB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 66633 bytes 9819551 (9.8 MB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

facu@facu-notebook:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=114 time=13.2 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=114 time=12.8 ms
```

En estas imágenes se puede ver como se le asigno la ip a la notebook (192.168.1.3) y que también tiene internet.

## 8. Defensa Oral

En la defensa oral del proyecto, se presentaron los resultados obtenidos durante la implementación de la Raspberry Pi como servidor DHCP y gateway para compartir la conexión a Internet con las notebooks conectadas a una red local. A lo largo de la defensa, el profesor realizó una serie de preguntas relacionadas con el proceso de configuración, los desafíos enfrentados y las soluciones implementadas. A continuación, se detallan las preguntas realizadas, los logros alcanzados y los problemas que quedaron pendientes.

### 8.1. Descripción de la Defensa Oral

Durante la defensa, el equipo expuso el proceso completo de configuración de la Raspberry Pi como servidor de red. Se mostró cómo se logró la automatización de la asignación de direcciones IP mediante el servidor DHCP y la implementación de NAT para compartir la conexión a Internet. También se discutieron las pruebas realizadas para verificar la conectividad y las soluciones a las preguntas planteadas por el profesor.

El profesor realizó preguntas específicas para evaluar los conocimientos sobre las configuraciones realizadas y la capacidad de resolver problemas en tiempo real, relacionados con la gestión de la red, seguridad y optimización de los recursos de red.

### 8.2. Preguntas Realizadas por el Profesor

#### 8.2.1. Pregunta 1: Medir transferencia de datos (Logrado)

El profesor pidió medir la cantidad de datos transferidos entre la Raspberry Pi y una de las notebooks conectadas. Esto se logró utilizando el comando iptables para monitorear los paquetes de red que pasaban por las interfaces de la Raspberry Pi. Se utilizó la siguiente regla para contabilizar los bytes transferidos:

```
sudo iptables -L -v -n
```

Esta salida permite ver la cantidad de datos transmitidos y recibidos en las interfaces de red.

#### **8.2.2. Pregunta 2: Bloquear una IP (Logrado)**

El profesor solicitó bloquear una dirección IP específica en la red. Esto se logró utilizando iptables para denegar el acceso de la IP indicada. El siguiente comando bloqueó todo el tráfico proveniente de la IP especificada:

```
sudo iptables -A INPUT -s 192.168.1.X -j DROP
```

La regla fue aplicada con éxito, impidiendo que la IP bloqueada accediera a la red.

#### **8.2.3. Pregunta 3: Revisar funcionamiento del DHCP (Logrado)**

El profesor pidió revisar el funcionamiento del servidor DHCP para confirmar que estaba asignando direcciones IP correctamente a las notebooks conectadas. Se verificó utilizando el comando:

```
sudo systemctl status isc-dhcp-server
```

Se confirmó que las direcciones IP estaban siendo asignadas dentro del rango definido en el archivo de configuración, lo que evidenció que el servidor DHCP estaba funcionando correctamente.

#### **8.2.4. Pregunta 4: Conexión a Internet a través del DHCP (Logrado)**

Se solicitó verificar que las notebooks conectadas al servidor DHCP también tuvieran acceso a Internet. Se realizó una prueba de conectividad con un ping a un servidor externo (por ejemplo, google.com) desde una notebook, confirmando que la conexión a Internet funcionaba correctamente a través de la Raspberry Pi.

#### **8.2.5. Pregunta 5: Bloquear una página web (Parcialmente Logrado)**

Se solicitó bloquear el acceso a un sitio web específico. Se intentó bloquear la página utilizando iptables con el siguiente comando:

```
sudo iptables -A OUTPUT -p tcp -d X.X.X.X -j REJECT
```

Sin embargo, el bloqueo fue incompleto debido a que algunos subdominios seguían siendo accesibles. Aunque la página principal fue bloqueada, los subdominios no pudieron ser controlados de manera efectiva en este intento, lo que dejó el resultado como semi logrado.

### **8.2.6. Pregunta 6: Limitar velocidad de transferencia de datos (No Logrado)**

El profesor pidió limitar la velocidad de transferencia de datos de una de las notebooks. A pesar de los intentos de aplicar limitaciones de velocidad utilizando iptables y otras herramientas, no se logró implementar esta funcionalidad de manera efectiva durante la defensa.

### **8.2.7. Pregunta 7: Guardar reglas de iptables (Logrado)**

El profesor solicitó verificar si las reglas de iptables eran persistentes después de un reinicio. Esto se logró utilizando el comando iptables-save para guardar las reglas actuales en un archivo de configuración:

```
sudo iptables-save > /etc/iptables/rules.v4
```

Este método aseguró que las reglas de firewall se mantuvieran activas después de reiniciar la Raspberry Pi.

### **8.2.8. Pregunta 8: Crear script de inicio para iptables (No Logrado)**

El profesor pidió crear un script que cargara automáticamente las reglas de iptables al iniciar el sistema. Aunque se intentó escribir un script en /etc/network/if-pre-up.d/iptables que ejecutara las reglas de firewall al inicio, no se logró implementar completamente esta funcionalidad durante la defensa.

### **8.2.9. Pregunta 9: Pasar de IPv4 a IPv6 (Parcialmente Logrado)**

El profesor solicitó que se configure el servidor DHCP para trabajar con direcciones IPv6.

#### Paso 1: Configuración del Reenvío de IPs para IPv6:

El primer paso fue habilitar el reenvío de IPs (IP forwarding) para IPv6, lo cual permite que la Raspberry Pi pueda reenviar paquetes IPv6 entre las interfaces de red. Para habilitarlo, se utilizó el siguiente comando:

```
sudo sysctl -w net.ipv6.conf.all.forwarding=1
```

Para hacer que este cambio sea permanente (para que no se pierda después de un reinicio), se editó el archivo **/etc/sysctl.conf** y se añadió la siguiente línea:

```
net.ipv6.conf.all.forwarding=1
```

#### Paso 2: Configuración del Servidor ISC DHCPv6:

Una vez instalado el servidor, se configuró el archivo de configuración **/etc/dhcp/dhcpd6.conf** para definir el rango de direcciones IPv6 que el servidor DHCP asignará a los dispositivos conectados. A continuación, se muestra un ejemplo de la configuración:

```
subnet6 2001:db8::/64 {  
    range6 2001:db8::10 2001:db8::100;  
    option dhcp6.name-servers 2001:4860:4860::8888,  
2001:4860:4860::8844; # Servidores DNS públicos de Google para IPv6  
    option dhcp6.domain-search "example.com";  
    option dhcp6.ia-pd 0 {  
        prefix 2001:db8:1::/48;  
        prefix-length 48;  
    };  
}
```

subnet6 define la red IPv6 que se está utilizando (en este caso, 2001:db8::/64).

range6 define el rango de direcciones IPv6 que el servidor asignará a los dispositivos.

option dhcp6.name-servers especifica los servidores DNS para IPv6 (en este caso, los DNS de Google).

option dhcp6.ia-pd define un prefijo delegable para los routers en la red, si es necesario.

#### Paso 3: Configuración de la Interfaz de Red para IPv6:

Se configuró el archivo **/etc/network/interfaces** para definir la interfaz de red que gestionaría el tráfico IPv6. La configuración fue similar a la que se utilizó para IPv4, pero añadiendo las opciones para IPv6. Aquí está un ejemplo:

```
auto eth0  
iface eth0 inet6 static  
    address 2001:db8::1  
    netmask 64
```

Esto asegura que la interfaz eth0 tenga una dirección IPv6 estática asignada, que se utilizará como puerta de enlace para los dispositivos de la red.

#### Paso 4: Reinicio del Servidor DHCPv6:

Después de configurar el archivo de configuración para IPv6, se reinició el servidor DHCP para aplicar los cambios:

```
sudo systemctl restart isc-dhcp-server
```

#### Paso 5: Verificación del Funcionamiento del DHCPv6:

Finalmente, se realizaron pruebas para verificar que los dispositivos conectados a la red recibieran direcciones IPv6 de manera adecuada. Para comprobar si las notebooks estaban obteniendo direcciones IPv6 correctamente, se utilizó el comando ip addr en las notebooks:

```
ip -6 addr show
```

Este comando mostró las direcciones IPv6 asignadas a las interfaces de red de los dispositivos.

#### Problemas Detectados:

A pesar de que la configuración general del servidor DHCPv6 fue implementada correctamente y que los dispositivos reciban direcciones IPv6 de manera adecuada. La conexión a internet se estaba enviando por protocolo IPv4, lo que llevó a que el resultado fuera parcialmente logrado.

#### **8.2.10. Pregunta 10: Conectar Wi-Fi (No Logrado)**

El profesor pidió que se configurara la conexión Wi-Fi en la Raspberry Pi para que actuara como una interfaz de red adicional. Aunque se intentó habilitar la conexión Wi-Fi y configurar la red a través del archivo **/etc/wpa\_supplicant/wpa\_supplicant.conf**, no se llegó, por motivos de tiempo, a establecer una conexión estable durante la defensa.

### **9. Conclusiones**

El proyecto de configuración de una Raspberry Pi como servidor DHCP y gateway para compartir la conexión a Internet en una red local ha permitido experimentar con diversas configuraciones de red, implementar tecnologías de gestión automatizada de direcciones IP y resolver problemas de conectividad dentro de un entorno controlado. A continuación, se presentan las conclusiones generales sobre el desarrollo del proyecto, los logros alcanzados, los desafíos superados y las posibles mejoras que podrían implementarse.

#### **9.1. Evaluación General del Proyecto**

El proyecto proporcionó una solución funcional para la gestión de una red local utilizando una Raspberry Pi como servidor central. Se logró automatizar la asignación de direcciones IP a través de ISC DHCP Server, lo que simplificó la administración de la red al eliminar la necesidad de configuraciones manuales de IP en cada dispositivo. Además, la implementación de NAT (Network Address Translation) permitió que todos los dispositivos conectados a la red local compartieran una única conexión a Internet de manera eficiente.

A pesar de algunos desafíos técnicos, el proyecto cumplió con su objetivo general de aprobar la primera parte del trabajo (defensa oral) y de crear una red local gestionada de manera centralizada, con acceso a Internet y configuraciones de red automatizadas. Se desarrollaron soluciones prácticas para resolver problemas comunes de conectividad y se mejoró la comprensión de conceptos clave relacionados con redes, como DHCP, NAT y la configuración de cortafuegos con iptables.