.UBAfiuba
FACULTAD DE INGENIERÍA

# Redes

## Protocolos Capa 4

## TCP / UDP

Ing. Marcelo E. Volpi

Ing. Lucas Giorgi

Ing. Vanesa Llasat

# EL DATAGRAMA IP

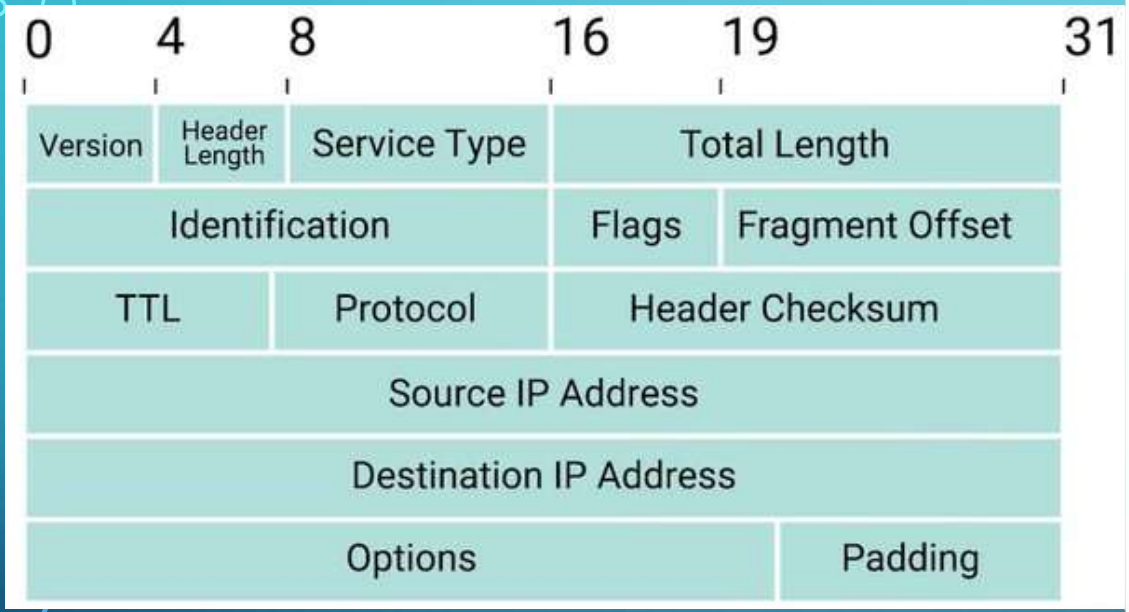
Internet es un conjunto de redes diferentes que comparten una pila de protocolos comunes. Cada una de estas redes es administrada por una entidad diferente: universidades, redes académicas nacionales, proveedores comerciales también llamados ISPs (Internet Service Providers), operadores, empresas multinacionales, etc.

Como consecuencia de esto las políticas de uso son muy variadas. Técnicamente a nivel de red la Internet puede definirse como un conjunto de redes o *sistemas autónomos* conectados entre sí que utilizan el protocolo de red IP. IP es una red de datagramas, no orientada a conexión, con servicio 'best effort', es decir no ofrece calidad de servicio o QoS (Quality of Service). La entrega de los paquetes no está garantizada ya que en momentos de congestión éstos pueden ser descartados sin previo aviso por los routers que se encuentren en el trayecto.

En una red IP la información viaja en datagramas IP. Esto incluye tanto el intercambio a nivel de transporte por TCP y UDP como cualquier información de control que tenga que intercambiarse, por ejemplo, para ruteo dinámico, mensajes de error, etc.

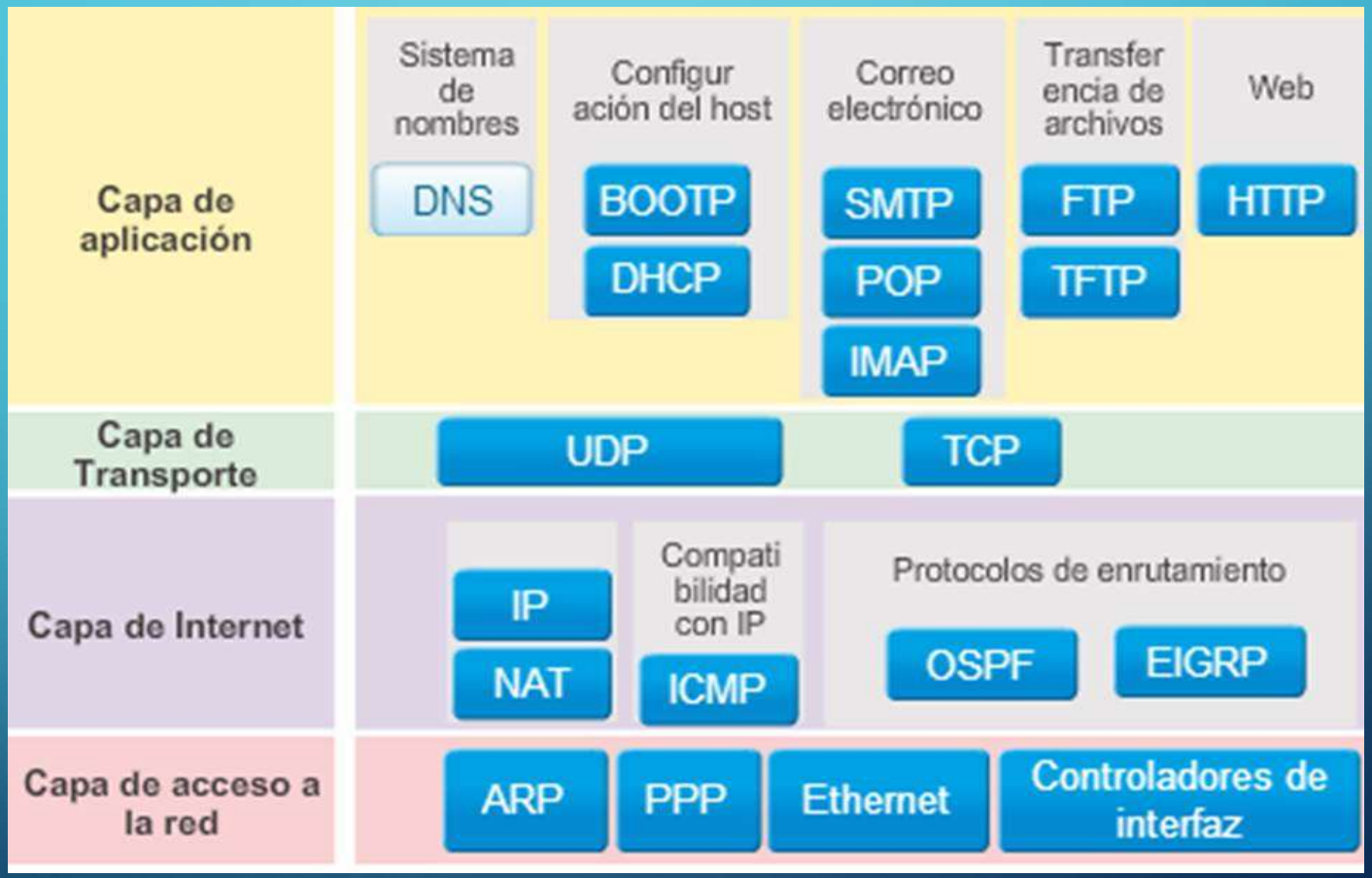
El datagrama IP tiene dos partes: cabecera y texto
La cabecera tiene una parte fija de 20 bytes y una opcional de entre 0 y 40 bytes. La longitud total de la cabecera siempre es múltiplo de 4; esto garantiza un proceso eficiente por parte de equipos (hosts o routers) cuya arquitectura optimiza el acceso a direcciones de memoria que estén en frontera de 32 bits.

# CABECERA IP V4




| Campo | Longitud (bits) |
|---|---|
| Versión | 4 |
| IHL (Internet Header Length) | 4 |
| DS (Differentiated Services) | 8 |
| Longitud total | 16 |
| Identificación | 16 |
| Reservado | 1 |
| DF (Don't Fragment) | 1 |
| MF (More Fragments) | 1 |
| Fragment offset | 13 |
| TTL (Time To Live) | 8 |
| Protocolo (de transporte) | 8 |
| Checksum (de cabecera) | 16 |
| Dirección de origen | 32 |
| Dirección de destino | 32 |
| Opciones | 0-320 |

# CABECERA IP V4



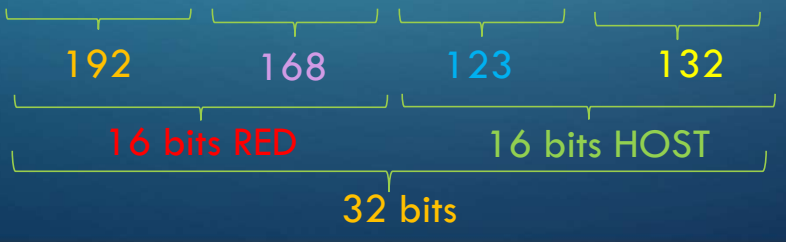
# DIRECCIONAMIENTO IP V4

Al configurar el protocolo TCP/IP en un dispositivo, las opciones de configuración de TCP/IP requieren lo siguiente:


## DIRECCION IP

**Una dirección IP es un número de 32 bits. Identifica de forma única un host (equipo u otro dispositivo, como una impresora o router) en una red TCP/IP.**
**Las direcciones IP normalmente se expresan en formato decimal punteado, con cuatro números separados por puntos, como 192.168.123.132.**

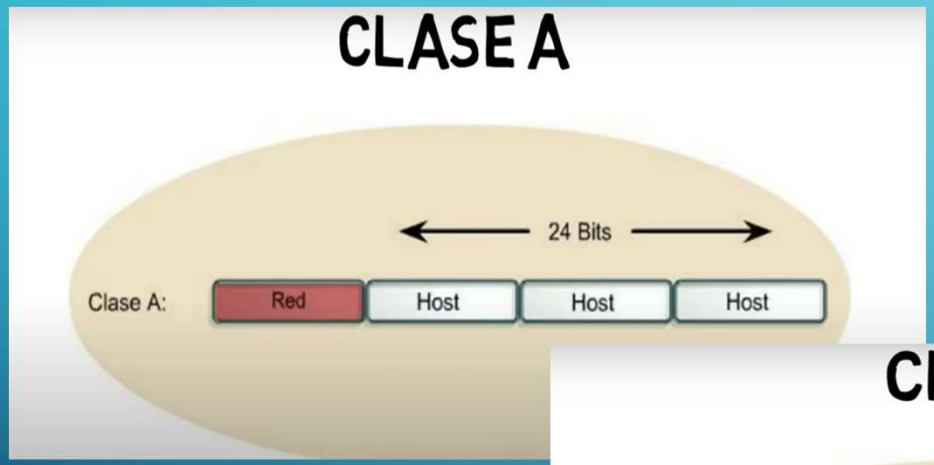
Por ejemplo, la dirección IP con puntos decimales 192.168.123.132 (32 bits en notación binaria
Numero de 32 bits: **11000000101010000111101110000100**

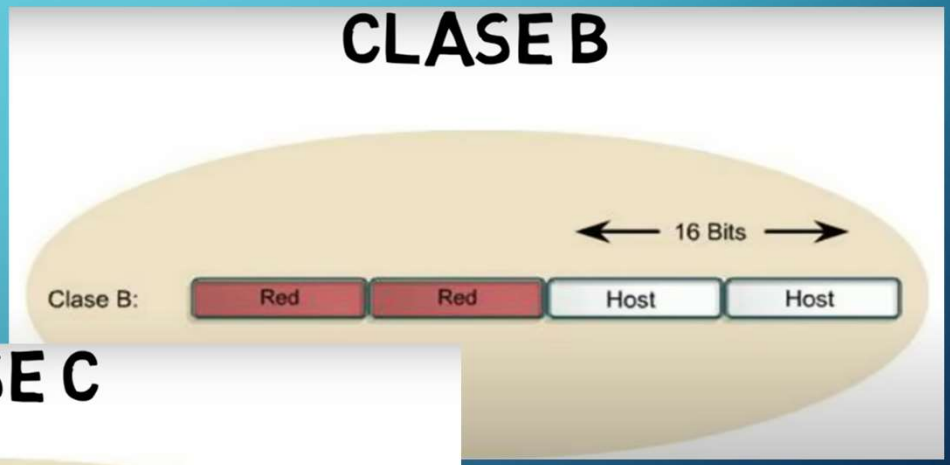


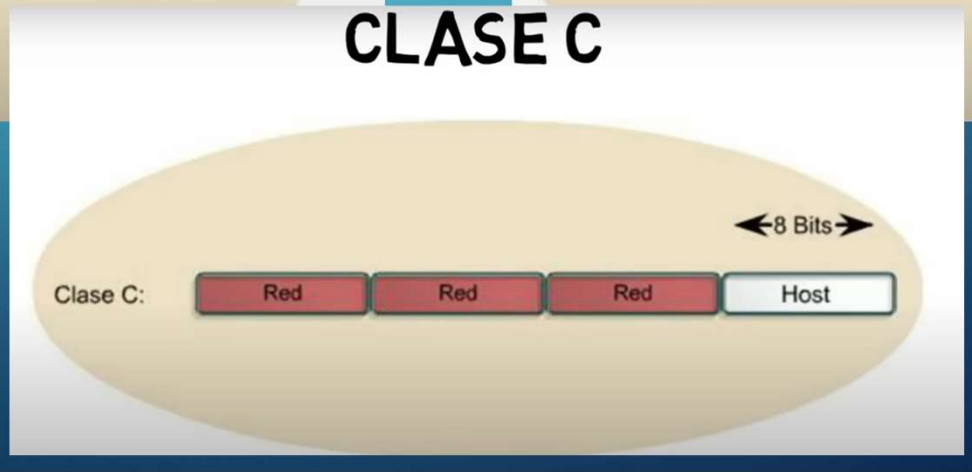
# CLASES DE DIRECCIONES IP

**EXISTEN 5 CLASES DE DIRECCIONES IP**

- **CLASE A** – SOPORTA GRANDES REDES EN INTERNET
- **CLASE B** – SOPORTA REDES EN INTERNET MODERADAS
- **CLASE C** - SOPORTA REDES PEQUEÑAS
- **CLASE D** - SOPORTA REDES MULTICAST
- **CLASE E** - SOLO EXPERIMENTAL



# COMO DETERMINAR LA CLASE DE UNA DIRECCIÓN IP

**CLASE A: EL PRIMER OCTETO ESTA COMPRENDIDO ENTRE 0-127** → 10.2.2.1

**CLASE B: EL PRIMER OCTETO ESTA COMPRENDIDO ENTRE 128-191** → 130.200.92.24

**CLASE C: EL PRIMER OCTETO ESTA COMPRENDIDO ENTRE 192-223** → 192.168.100.50

**CLASE D: EL PRIMER OCTETO ESTA COMPRENDIDO ENTRE 224-239**

**CLASE E: EL PRIMER OCTETO ESTA COMPRENDIDO ENTRE 240-255**

EJEMPLO CLASE C

192.168.100.50 / 24

24 bits RED

8 bits HOST

# MASCARA DE RED

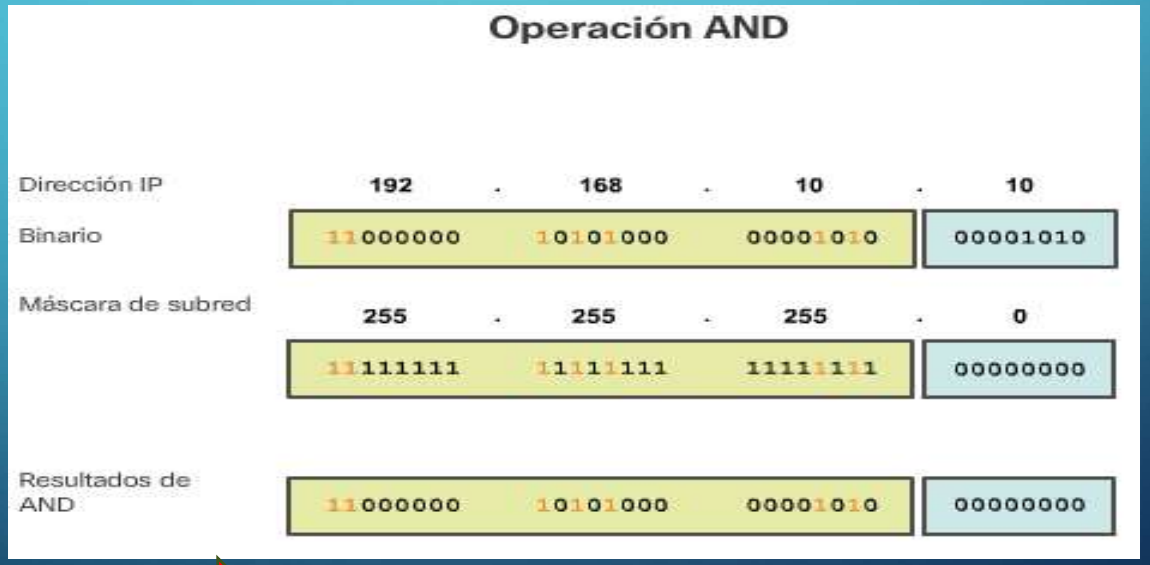
•Máscara de subred: se usa para identificar la porción de red/host de la dirección IPv4.

Para identificar la dirección de red de un host IPv4, se recurre a la operación lógica AND para la dirección IPv4, bit por bit, con la máscara de subred. El uso de la operación AND entre la dirección y la máscara de subred produce la dirección de red.

| Máscara de subred | Dirección de 32 bits | Longitud de prefijo |
|---|---|---|
| 255.0.0.0 | 11111111.00000000.00000000.00000000 | /8 |
| 255.255.0.0 | 11111111.11111111.00000000.00000000 | /16 |
| 255.255.255.0 | 11111111.11111111.11111111.00000000 | /24 |
| 255.255.255.128 | 11111111.11111111.11111111.10000000 | /25 |
| 255.255.255.192 | 11111111.11111111.11111111.11000000 | /26 |
| 255.255.255.224 | 11111111.11111111.11111111.11100000 | /27 |
| 255.255.255.240 | 11111111.11111111.11111111.11110000 | /28 |
| 255.255.255.248 | 11111111.11111111.11111111.11111000 | /29 |
| 255.255.255.252 | 11111111.11111111.11111111.11111100 | /30 |

1 AND 1 = 1
0 AND 1 = 0
0 AND 0 = 0
1 AND 0 = 0

# DETERMINACIÓN MASCARA DE RED

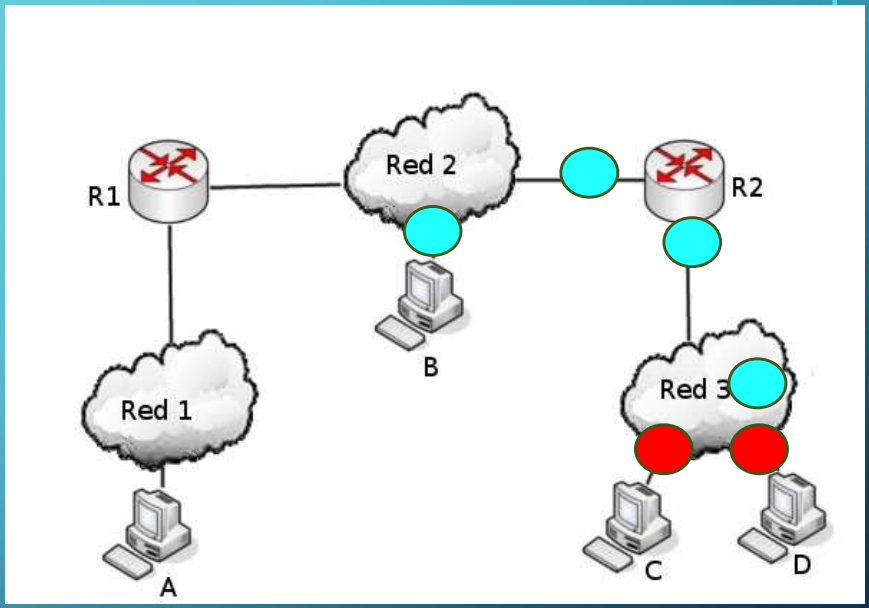




Dirección de Red Resultante 192 168 10 0

# EJEMPLO BASICO ENRUTAMIENTO DE PAQUETES

•Gateway predeterminado: identifica el gateway local (es decir, la dirección IPv4 de interfaz de router local) para llegar a redes remotas.

| Host | Red | Dirección IP | Dirección física |
|---|---|---|---|
| A | Red 1 | 192.168.0.10 | 00-60-52-0B-B7-7D |
| R1 | | 192.168.0.1 | 00-E0-4C-AB-9A-FF |
| | Red 2 | 10.10.0.1 | A3-BB-05-17-29-D0 |
| B | | 10.10.0.7 | 00-E0-4C-33-79-AF |
| R2 | | 10.10.0.2 | B2-42-52-12-37-BE |
| | Red 3 | 200.3.107.1 | 00-E0-89-AB-12-92 |
| C | | 200.3.107.73 | A3-BB-08-10-DA-DB |
| D | | 200.3.107.200 | B2-AB-31-07-12-93 |




En el ejemplo anterior, supongamos que el ordenador 200.3.107.200 (D) envía un mensaje al ordenador con 200.3.107.73 (C). Como ambas direcciones comienzan con los mismos números, D sabrá que ese ordenador se encuentra dentro de su propia red y el mensaje se entregará de forma directa. Sin embargo, si el ordenador 200.3.107.200 (D) tuviese que comunicarse con 10.10.0.7 (B), D advertiría que el ordenador destino no pertenece a su propia red y enviaría el mensaje al router R2 (es el ordenador que le da salida a otras redes). El router entregaría el mensaje de forma directa porque B se encuentra dentro de una de sus redes (la Red 2).

# ¿QUÉ ES UDP?

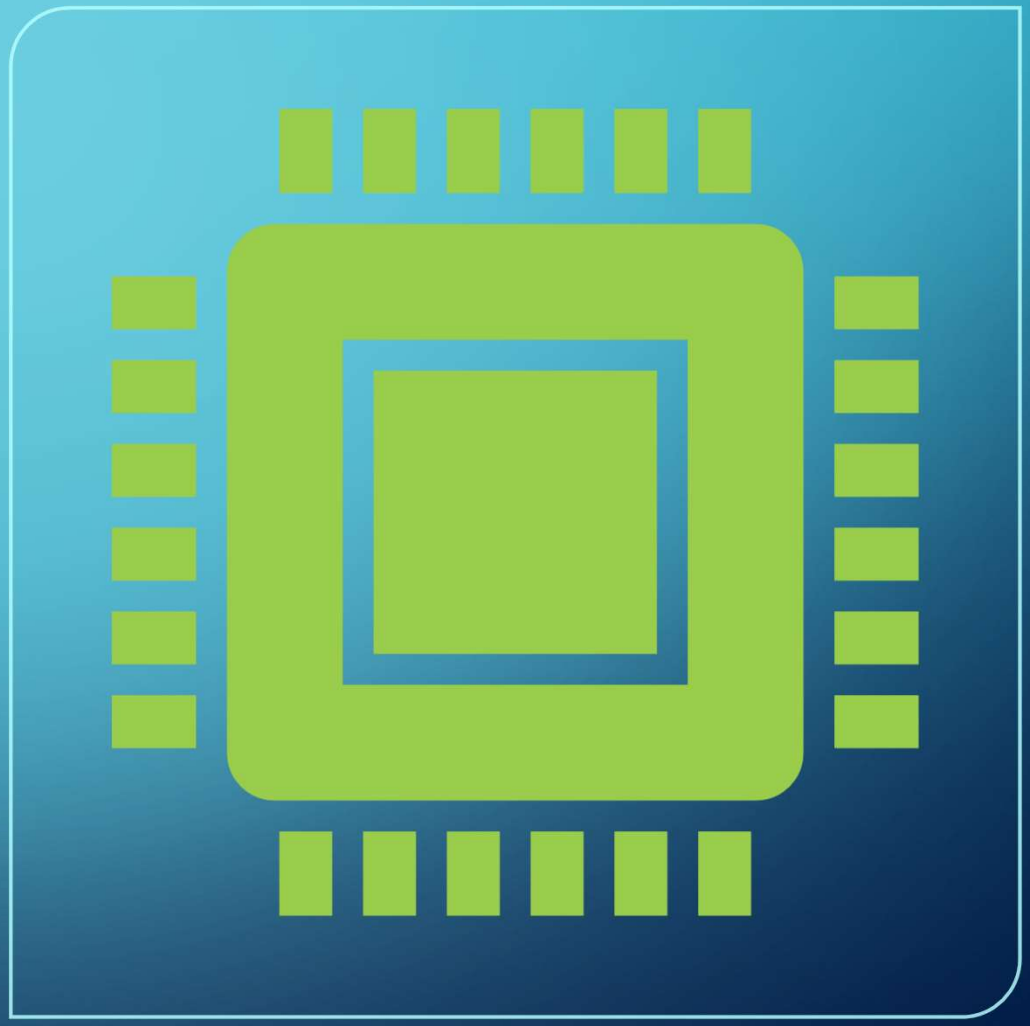
- UDP (**User Datagram Protocol o Protocolo de Datagramas de Usuario**) es uno de los protocolos principales de la capa de transporte del modelo OSI y del modelo TCP/IP. Fue diseñado para proporcionar una manera simple y eficiente de enviar datagramas a través de la red sin establecer una conexión previa ni realizar controles complejos de errores.

# FORMATO DEL ENCABEZADO UDP

- **Puerto de origen (16 bits)**
- Indica el puerto del emisor.
- **Puerto de destino (16 bits)**
- Indica el puerto del receptor
- **Longitud (16 bits):**
- Especifica el tamaño total del segmento UDP
- **Checksum (16 bits):**
- Valor de verificación que permite detectar errores simples en el encabezado y los datos

# PUERTOS EN UDP

- Los puertos UDP son números de 16 bits que van del 0 al 65535, divididos en tres rangos:
  - **Puertos bien conocidos** (0-1023): Reservados para servicios del sistema.
  - **Puertos registrados** (1024-49151): Asignados por la IANA para aplicaciones específicas.
  - **Puertos dinámicos/privados** (49152-65535): Disponibles para uso temporal.



# PROCESO DE COMUNICACIÓN

1. **Preparación inicial**: La aplicación emisora crea un socket UDP.
2. **Encapsulación del mensaje**: Los datos de la aplicación se encapsulan en un datagrama UDP.
3. **Envío al nivel de red**: El datagrama UDP se pasa al protocolo IP.
4. **Transporte por la red**: El paquete viaja por la red usando el enrutamiento IP.
5. **Recepción en el destino**: El host destino recibe el paquete IP.
6. **Entrega a la aplicación**: UDP identifica la aplicación destino usando el puerto.

# DIFERENCIAS CON TCP

| Factor | TCP | UDP |
|---|---|---|
| **Tipo de conexión** | Requiere una conexión establecida antes de transmitir datos | No se necesita conexión para iniciar y finalizar una transferencia de datos |
| **Secuencia de datos** | Puede secuenciar datos (enviar en un orden específico) | No puede secuenciar y ordenar datos |
| **Retransmisión de datos** | Puede transmitir datos si no llegan los paquetes | Sin transmisión de datos. Los datos perdidos no se pueden recuperar |
| **Entrega** | Garantizada | No garantizada |
| **Velocidad** | Lenta, pero entrega los datos completos | Rápida, pero existe el riesgo de que los datos se entreguen incompletos |

# VENTAJAS DEL PROTOCOLO UDP

- **Baja latencia**

  Sin Handshake.

- **Menor sobrecarga**

  Cabecera simple. Sin estado de conexión. Sin control de flujo.

- **Simplicidad**

  Fácil implementación. Menos código. No hay limitaciones debido al diseño del protocolo.

# DESVENTAJAS Y LIMITACIONES DEL PROTOCOLO UDP

- **Falta de control de congestión**

  Puede contribuir a la congestión. Debe implementarlo el cliente si es necesario.

- **Sin garantía de entrega**

  Modelo best-effort. Limita los usos.

- **Sin ordenamiento de paquetes**

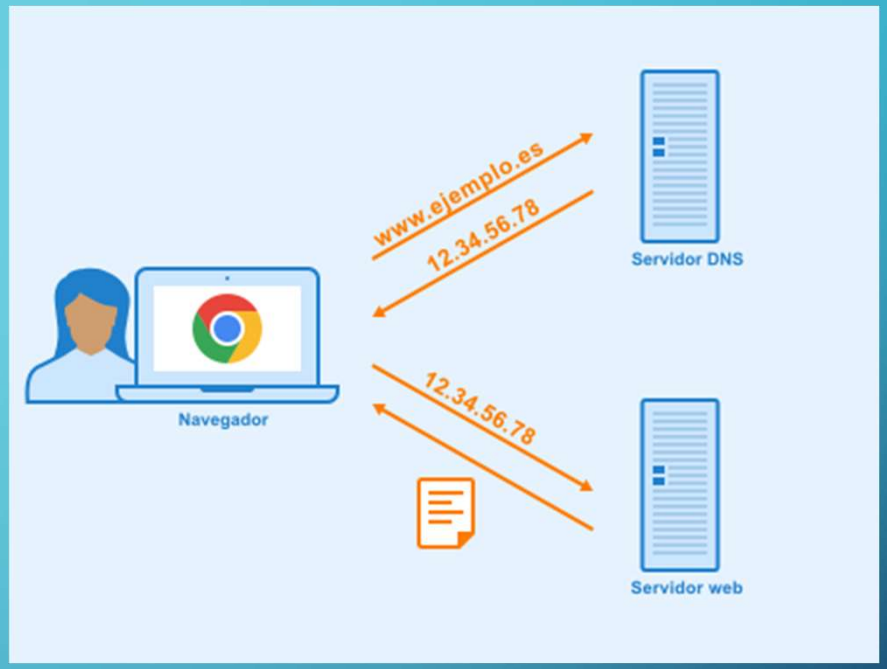
  Puede complicar la detección de datos perdidos. Debe implementarlo el cliente si es necesario.

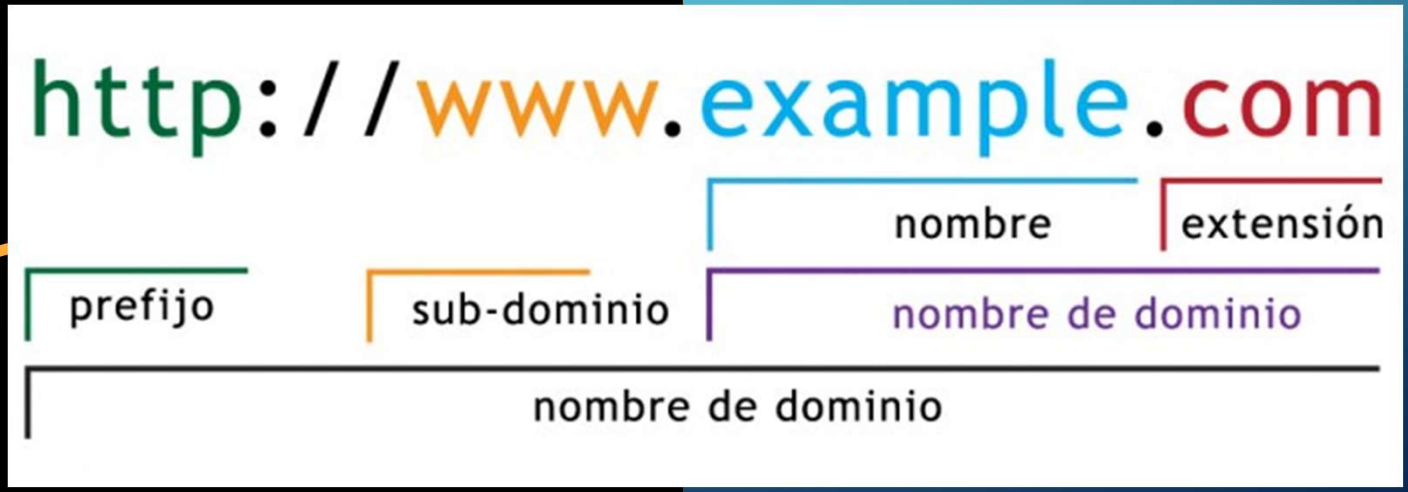
# CASOS DE USO

- **VoIP**
- **Juegos en Línea**
- **Streaming de Video**
- **DNS**
- **NTP**

# SISTEMA DE NOMBRES DE DOMINIO

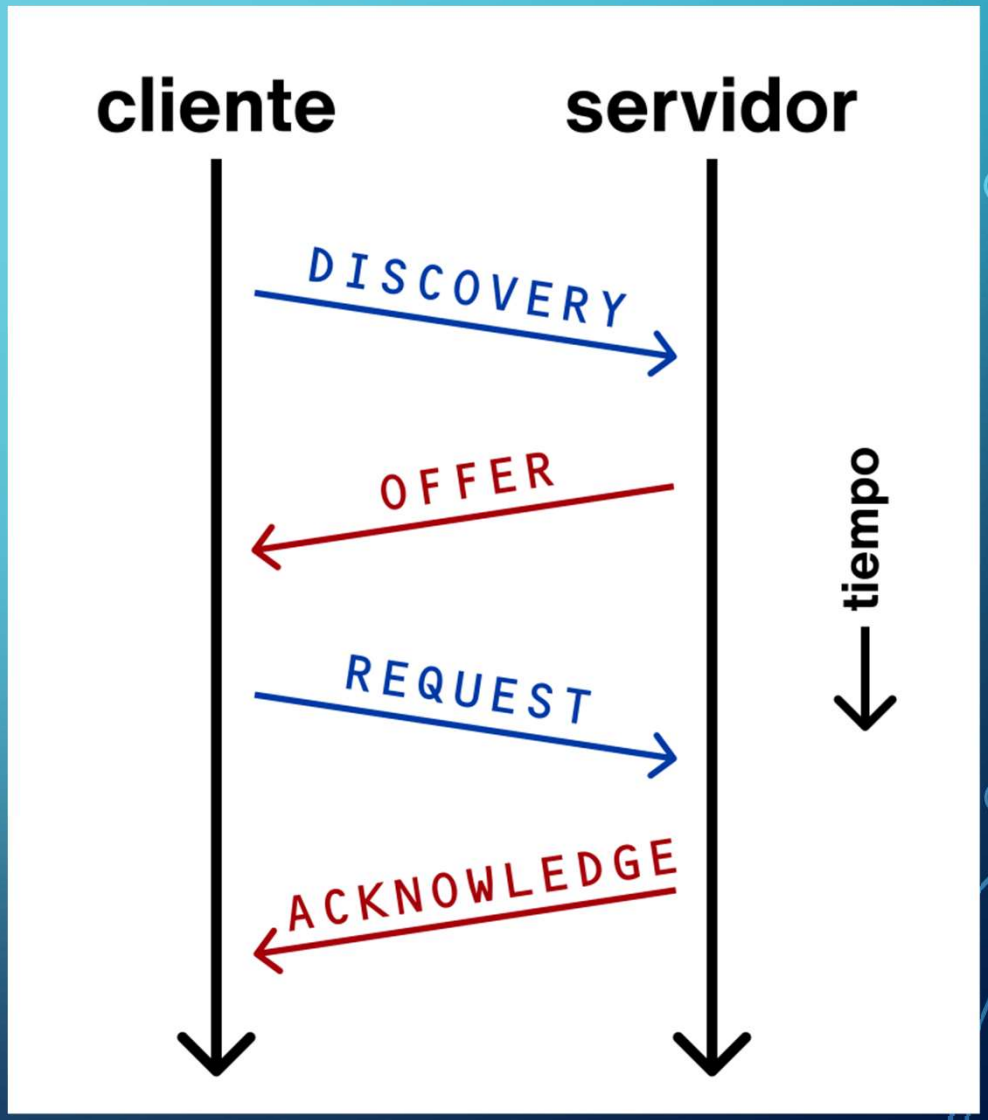
## DNS - DOMAIN NAME SYSTEM



# DHCP

# PROTOCOLO DE CONFIGURACIÓN DINÁMICA DE HOST



# SEGURIDAD EN UDP

- Vulnerabilidades
- Protecciones esenciales
- UDP en redes modernas
- El futuro de UDP

# VULNERABILIDADES COMUNES EN UDP

## Ataques de Inundación UDP

Envío masivo de datagramas UDP para saturar el ancho de banda y agotar los recursos del servidor, causando denegación de servicio (DoS).

## Desbordamiento de Búfer

El procesamiento de datagramas UDP maliciosos puede generar errores de memoria, permitiendo la ejecución remota de código en ciertas implementaciones.

## Ataques de Reflexión

El atacante falsifica la IP de la víctima, enviando datagramas a servidores legítimos que responden a la víctima, amplificando el tráfico y provocando DoS.

## Ataques Loop DoS

Creación de bucles de tráfico entre dispositivos mal configurados, agotando sus recursos y causando interrupciones en la red.

# MEDIDAS DE PROTECCIÓN ESENCIALES

## Firewalls e IPS

Bloquean tráfico no deseado y sospechoso. Es crucial configurar reglas para limitar el acceso a puertos UDP no utilizados, fortaleciendo la seguridad perimetral.

## Rate Limiting

Restringe la cantidad de paquetes recibidos de una fuente en un tiempo dado, reduciendo la posibilidad de sobrecarga y ataques de inundación.

## Validación de Datos

Los servidores deben validar rigurosamente los datagramas entrantes, asegurando que cumplan con los formatos esperados y no contengan datos maliciosos.

## Monitoreo y Detección

Herramientas de monitoreo de red y sistemas de detección de intrusiones identifican patrones de tráfico anómalos, alertando sobre posibles ataques en curso.

# UDP EN REDES MODERNAS: INNOVACIÓN Y FUTURO

## QUIC: Velocidad y Seguridad

Protocolo de Google que combina UDP con fiabilidad y cifrado TLS 1.3, reduciendo la latencia y evitando ataques comunes. Es la base de HTTP/3.

## DTLS: TLS sobre UDP

Una versión de TLS adaptada para UDP, proporcionando seguridad a aplicaciones que requieren baja latencia, como la comunicación en tiempo real.

## RTP: Streaming en Tiempo Real

Utilizando en streaming de audio y video, RTP aprovecha la velocidad de UDP para transmisiones en tiempo real , esencial para telefonía IP y videoconferencias.

# EL FUTURO PROMETEDOR DE UDP

## Crecimiento de Aplicaciones en Tiempo Real

La demanda de conexiones rápidas y seguras impulsa la renovación de UDP.

## Flexibilidad y Velocidad

A pesar de sus debilidades, su diseño lo convierte en una base ideal para nuevas soluciones.

## Extensión de Capacidades

Desarrollo de protocolos que añaden seguridad y fiabilidad sin sacrificar baja latencia.

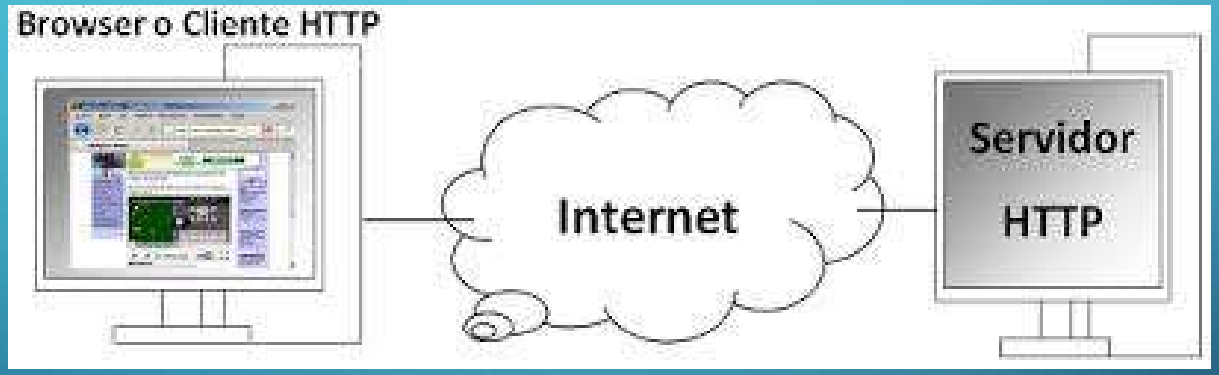
## Centro de la Evolución de Internet

Su uso en QUIC y HTTP/3 demuestra su vigencia y relevancia futura.

# HTTP (PROTOCOLO DE TRANSFERENCIA DE HIPERTEXTO)

La WWW (World Wide Web) es un conjunto de aplicaciones que se comunican a través del protocolo web conocido como Protocolo de transferencia de hipertexto (HTTP).

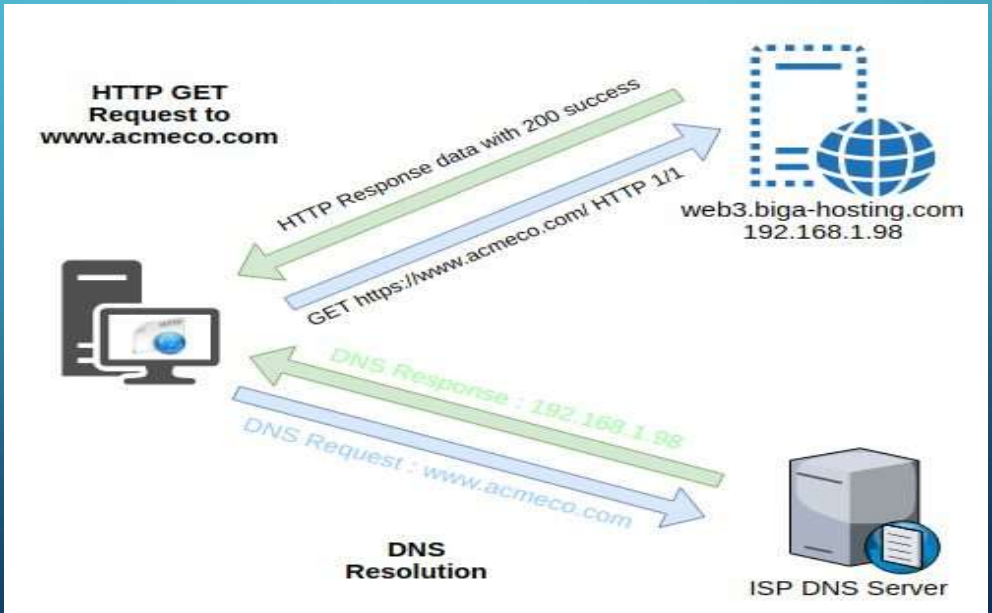
Existen navegadores web, aplicaciones web móviles y servidores web que se comunican a través de HTTP.




HTTP es un protocolo web (Capa de Aplicación OSI) que es una de las piedras fundamentales de cómo funciona Internet. Cuando visita un sitio web, HTTP se utiliza para entregar el contenido de esa página, mostrándolo en su navegador. El protocolo web, HTTP, es un protocolo de solicitud-respuesta que define cómo se comunican los clientes web con los servidores web.

# HTTP (PROTOCOLO DE TRANSFERENCIA DE HIPERTEXTO)

Pasos de un flujo de solicitud-respuesta HTTP:
El flujo de solicitud-respuesta del protocolo HTTP se produce de la siguiente manera.
Cuando un usuario o una aplicación navega a un sitio web, por ejemplo, www.google.com, vemos casi instantáneamente una imagen en el navegador o la aplicación que muestra el contenido esperado del sitio web. Sin embargo, bajo la superficie, a menudo se hacen cientos de solicitudes y se devuelven las respuestas.
Un navegador web es un ejemplo de cliente web, pero también lo son las aplicaciones web móviles que probablemente tenga en su dispositivo iPhone o Android. Sin embargo, para mostrar cómo se realizan estas solicitudes y respuestas en el back-end cuando alguien navega a un sitio web, el proceso se puede dividir en cuatro pasos:



# HTTP (PROTOCOLO DE TRANSFERENCIA DE HIPERTEXTO)

Paso 1: Navegación e iniciación
El usuario escribe una dirección web en un navegador o hace clic en un enlace de un correo electrónico u otra comunicación. La URL también contiene el dominio. El navegador busca la dirección web con una búsqueda de DNS y, a continuación, envía la solicitud a esta dirección.

Paso 2: El cliente envía un mensaje de solicitud HTTP al servidor
El cliente HTTP, por ejemplo, el navegador, construye un mensaje de solicitud que se dirige al servidor web de Google. La primera línea del mensaje de solicitud de HTTP identifica la página raíz del sitio web, por ejemplo, get /.
Esta línea indica la versión de HTTP, por ejemplo, la versión HTTP 1.1 o HTTP 1.0. Después de esta línea inicial, otra serie denominada "encabezados de solicitud" proporciona información adicional sobre la solicitud e información sobre la entidad solicitante, por ejemplo, el navegador. Una vez que el mensaje de solicitud se envía al servidor web, se puede leer y crear una respuesta.

Paso 3: El servidor web de Google envía la respuesta HTTP de vuelta al cliente
Una vez que el servidor web de Apple recibe una solicitud, se crea un mensaje de respuesta y se devuelve al navegador (cliente). La primera línea del mensaje incluye el código de respuesta "200 OK" para indicar que el servidor web podría responder a la solicitud correctamente.
Otros códigos de respuesta incluyen:
- 404 — No encontrado
- 502 — Puerta de enlace no válida
- 503 — Servicios no disponibles

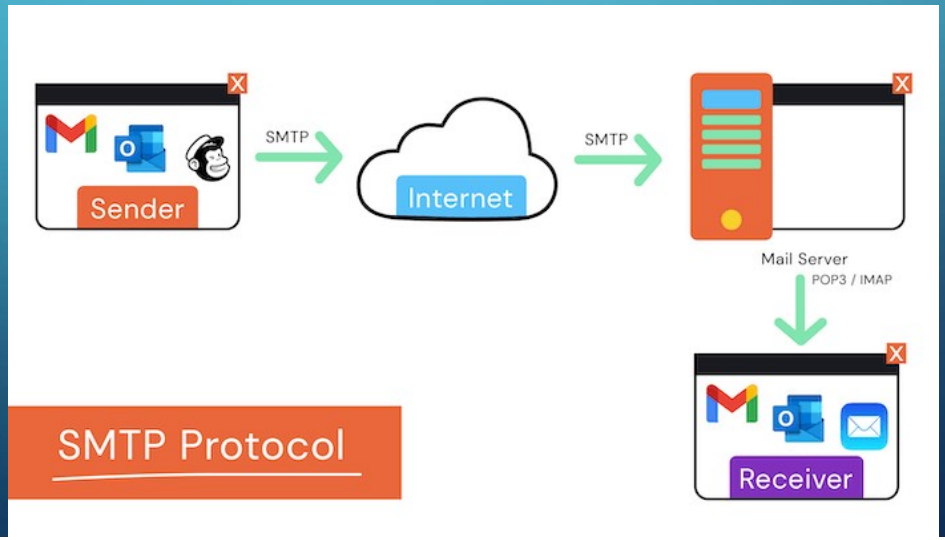
El formato exacto del mensaje de solicitud se repite en el mensaje de respuesta HTTP con una serie de líneas llamadas "encabezados de respuesta", que proporcionan información sobre la respuesta. Después de los encabezados de respuesta aparece una línea en blanco seguida de la página web real, en forma de documento presentado mediante el lenguaje de marcado de hipertexto HTML.
Una vez generado el mensaje de respuesta HTTP, el servidor web envía el mensaje de vuelta al navegador y el navegador recibe y lee la respuesta.

Paso 4: Mensaje representado por el explorador
El paso final es que el navegador representa el mensaje de respuesta y muestra la página web de Apple en el navegador.

# SMTP (PROTOCOLO DE TRANSFERENCIA SIMPLE DE CORREO)

Es un estándar técnico para la transmisión de correo electrónico a través de una red. Al igual que otros protocolos de red, SMTP permite a los ordenadores (computadoras) y servidores intercambiar datos independientemente de su hardware o software subyacente. Al igual que el uso de una forma estandarizada de escribir una dirección en un sobre permite el funcionamiento del servicio postal, el protocolo SMTP estandariza la forma en que el correo electrónico viaja del remitente al destinatario, permitiendo la entrega generalizada de correo electrónico.

SMTP es un protocolo de entrega de correo, no un protocolo de recuperación de correo. Un servicio postal entrega el correo en un buzón, pero el destinatario tiene que recuperar el correo del buzón. Del mismo modo, el protocolo SMTP entrega un correo electrónico al servidor de correo de un proveedor de correo electrónico, pero se utilizan protocolos independientes para recuperarlo del servidor de correo para que el destinatario pueda leerlo.



# FUNCIONAMIENTO DE SMTP

Todos los protocolos de red siguen un proceso predefinido para el intercambio de datos. SMTP define un proceso para el intercambio de datos entre un cliente de correo electrónico y un servidor de correo. Un cliente de correo electrónico es con lo que interactúa un usuario: el ordenador (computadora) o la aplicación web donde accede y envía los correos electrónicos. Un servidor de correo es un ordenador o computadora especializada en enviar, recibir y reenviar correos electrónicos. Los usuarios no interactúan directamente con los servidores de correo.

En resumen se describe lo que pasa entre el cliente de correo electrónico y el servidor de correo para que comience el envío:

Apertura de la conexión SMTP: dado que SMTP utiliza el protocolo de control de transmisión (TCP) como protocolo de transporte, este primer paso comienza con una conexión TCP entre el cliente y el servidor. A continuación, el cliente de correo electrónico comienza el proceso de envío del correo con un comando especializado "Hello" (HELO o EHLO, descrito más adelante).

Transferencia de los datos del correo electrónico: el cliente envía al servidor una serie de comandos acompañados del contenido real del correo electrónico. El encabezado del correo (incluido el destino y el asunto), el cuerpo del correo y cualquier otro elemento.

Agente de transferencia de correo (MTA): el servidor ejecuta un programa llamado agente de transferencia de correo (MTA). El MTA comprueba el dominio de la dirección de correo electrónico del destinatario y, si difiere de la del remitente, consulta el sistema de nombres de dominio (DNS) para encontrar la dirección IP del destinatario. Es como si una oficina de correos buscara el código postal del destinatario del correo.

Cierre de la conexión: el cliente avisa al servidor cuando la transmisión de datos ha terminado, y el servidor cierra la conexión. En este punto, el servidor no recibirá más datos de correo electrónico del cliente a menos que éste abra una nueva conexión SMTP.

# ELEMENTOS DE SMTP

¿Qué es un sobre SMTP?
El "sobre" SMTP es el conjunto de información que el cliente de correo electrónico envía al servidor de correo sobre la procedencia del correo y su destino. El sobre SMTP es distinto del encabezado y el cuerpo del correo electrónico y no es visible para el destinatario del mismo.

¿Qué son los comandos SMTP?
Son comandos predefinidos basados en texto que indican al cliente o al servidor qué hacer y cómo gestionar los datos adjuntos. Piensa en ellos como botones que el cliente puede pulsar para que el servidor acepte los datos correctamente.

HELO/EHLO: estos comandos dicen "Hello" e inician la conexión SMTP entre el cliente y el servidor. "HELO" es la versión básica de este comando. "EHLO" es para un tipo especializado de SMTP.
MAIL FROM: indica al servidor quién envía el correo electrónico. Si Carlos intentara enviar un correo electrónico a su amigo Bob, un cliente podría enviar "MAIL FROM: "<carlos@example.com>".
RCPT TO: este comando sirve para enumerar los destinatarios del correo electrónico. Un cliente puede enviar este comando varias veces si hay varios destinatarios. En el ejemplo anterior, el cliente de correo electrónico de Carlos enviaría "RCPT TO: <bob@example.com>".

# PUERTOS DE SMTP

¿Qué puerto utiliza SMTP?

En redes, un puerto es el punto virtual en el que se reciben los datos de la red. Piensa en él como el número de piso en la dirección de una carta. Los puertos ayudan a los ordenadores o computadoras a clasificar los datos de red en las aplicaciones correctas. Las medidas de seguridad de red, como los firewalls, pueden bloquear los puertos innecesarios para evitar el envío y la recepción de datos malintencionados.

En el pasado, SMTP solo utilizaba el puerto 25. Hoy en día, SMTP sigue utilizando el puerto 25, pero también puede utilizar los puertos 465, 587 y 2525.

- El puerto 25 es el más utilizado para las conexiones entre servidores SMTP. En la actualidad, los firewalls de las redes de usuarios finales suelen bloquear este puerto, ya que los servidores de correo no deseado intentan abusar de él para enviar grandes cantidades de correo no deseado.
- El puerto 465 fue designado en su día para el uso de SMTP con encriptación de Secure Sockets Layer (SSL). Pero SSL fue sustituido por Transport Layer Security (TLS), y los sistemas de correo electrónico modernos no utilizan este puerto. Solo aparece en los sistemas heredados (obsoletos).
- El puerto 587 es ahora el puerto por defecto para el envío de correo electrónico. Las comunicaciones SMTP a través de este puerto utilizan la encriptación TLS.
- El puerto 2525 no está oficialmente asociado a SMTP, pero algunos servicios de correo electrónico ofrecen el envío de SMTP a través de este puerto en caso de que los puertos anteriores estén bloqueados.