

Glosario Técnico: Protocolos de Comunicación

TCP/IP

1. Protocolo de Comunicación

Conjunto de reglas que determinan cómo se transmiten, reciben e interpretan los datos en una red. Incluye normas de sintaxis, semántica y temporización.

2. Modelo TCP/IP

Arquitectura en capas que define cómo se comunican los dispositivos en Internet y redes privadas. Se compone de 4 capas: Aplicación, Transporte, Internet y Enlace.

3. Capa de Aplicación (TCP/IP)

Nivel más alto del modelo TCP/IP, donde los usuarios y aplicaciones acceden a los servicios de red mediante protocolos como HTTP, FTP, SMTP, y DNS.

4. HTTP (HyperText Transfer Protocol)

Protocolo de la capa de aplicación utilizado para la navegación web. Su versión segura, HTTPS, incluye cifrado mediante SSL/TLS.

5. HTTPS (HyperText Transfer Protocol Secure)

Versión cifrada de HTTP que protege la confidencialidad e integridad de los datos mediante protocolos criptográficos como TLS.

6. FTP (File Transfer Protocol)

Protocolo que permite la transferencia de archivos entre un cliente y un servidor. No cifra los datos, lo que representa un riesgo de seguridad.

7. SMTP (Simple Mail Transfer Protocol)

Protocolo utilizado para el envío de correos electrónicos entre servidores en la capa de aplicación.

8. DNS (Domain Name System)

Servicio de red que convierte nombres de dominio legibles (ej. www.ejemplo.com) en direcciones IP.

9. Capa de Transporte (TCP/IP)

Encargada de la comunicación extremo a extremo entre dispositivos. Utiliza protocolos como TCP y UDP para gestionar la entrega de datos.

10. TCP (Transmission Control Protocol)

Protocolo de transporte orientado a conexión que garantiza la entrega ordenada y sin errores mediante mecanismos como control de flujo y control de congestión.

11. UDP (User Datagram Protocol)

Protocolo de transporte no orientado a conexión, más rápido pero sin garantías de entrega. Se usa en aplicaciones como streaming y juegos en línea.

12. Control de Flujo

Mecanismo del protocolo TCP que regula la cantidad de datos que puede enviar el emisor según la capacidad del receptor.

13. Control de Congestión

Técnica de TCP que ajusta la tasa de envío de datos para evitar la saturación de la red. Utiliza algoritmos como *Slow Start* y *Congestion Avoidance*.

14. Capa de Red / Internet (TCP/IP)

Encargada de enrutar paquetes desde el origen al destino a través de múltiples redes. Utiliza protocolos como IP e ICMP.

15. IP (Internet Protocol)

Protocolo que asigna direcciones lógicas únicas a cada dispositivo y permite el encaminamiento de paquetes a través de redes.

16. ICMP (Internet Control Message Protocol)

Protocolo usado para el diagnóstico y control de errores en redes. Es la base de herramientas como **ping**.

17. Capa de Enlace (TCP/IP)

Nivel que conecta directamente los dispositivos dentro de una red local. Se encarga del direccionamiento físico y la entrega de tramas.

18. Ethernet

Tecnología de red que define cómo los dispositivos se comunican en una red LAN usando medios cableados.

19. ARP (Address Resolution Protocol)

Protocolo que convierte direcciones IP en direcciones MAC para permitir la entrega de paquetes dentro de una red local.

20. Encapsulación

Proceso mediante el cual los datos generados por una aplicación se preparan para ser transmitidos por la red, agregando encabezados en cada capa del modelo.

21. Wireshark

Herramienta gráfica de análisis de tráfico de red que permite capturar, visualizar y estudiar los paquetes que circulan por una red.

22. Tcpdump

Herramienta de línea de comandos para capturar y analizar tráfico de red en tiempo real. Útil para auditorías técnicas y análisis avanzados.

23. Segmentación (en TCP)

División de los datos de una aplicación en unidades más pequeñas (segmentos) para su transmisión por la red, con numeración para reensamblaje.

24. Desencapsulación

Proceso inverso a la encapsulación, en el cual un paquete recibido pasa por las capas del modelo hasta ser procesado por la aplicación destino.

25. Puerto (de red)

Número lógico que identifica servicios específicos dentro de un host. TCP y UDP usan puertos para dirigir los datos al servicio correcto (ej. HTTP = 80).
