



# Cuestionario 2

## “Introduction to Networking”

Nombre - Gerardo Antonio García Vázquez

Grupo - GIR0541

No. De control - 1222100818

Programación de Redes

Unidad III



**1. ¿Por qué algunos dispositivos funcionan en unas redes y otros no?**

La falta de una tarjeta de interfaz de red (NIC) impide la conexión de algunos dispositivos a la red. Y es necesario asegurarse de que las direcciones IP en la red y en los dispositivos estén en el mismo segmento.

**2. ¿Cómo se encuentran los dispositivos entre sí?**

Los dispositivos se comunican a través de un switch y un router. La computadora envía datos a la puerta de enlace. El switch busca la dirección MAC del router, el router busca la dirección MAC del destino y luego los datos se envían.

**3. ¿Qué son conmutadores, enrutadores e interfaces, y cómo se configuran?**

**Conmutadores (Switches):**

Son dispositivos de Capa 2 que conectan dispositivos en una red local (LAN).

Configuración: Conectan puertos RJ45 a otros dispositivos, Interfaces configurables, Utilizan ARP para determinar intercambio de datos, Microsegmentación mejora flujo de tráfico, Pueden configurarse para VLANs.

**Enrutadores (Routers):**

Son dispositivos de Capa 3 que facilitan la comunicación entre redes.

Configuración: Conectan puertos RJ45 a otros enrutadores o conmutadores, Interfaces configurables con direcciones IPv4, Utilizan tabla de enrutamiento, Permiten configuración de funciones adicionales.

**Interfaces:**

Son puntos de conexión físicos en conmutadores y enrutadores.

Configuración: Puertos RJ45 con al menos una interfaz configurable, Configuración mediante terminal o interfaz de línea de comandos, Creación de interfaces virtuales, puertos troncales y aplicar políticas.

**4. ¿Cómo se relacionan REST y SDN con esta información?**

SDN y REST están interrelacionados en el ámbito de la configuración dinámica de redes.

SDN utiliza el software para configurar y controlar dispositivos de red, mientras que REST proporciona un protocolo simplificado para acceder y manipular las funciones de estos dispositivos a través de API, facilitando la comunicación y configuración dinámica en un entorno SDN.

**5. Si un Switch es más eficiente que un Hub, ¿por qué comprarías un Hub?**

Aunque un Switch es más eficiente, un Hub permite la conexión simultánea de varios dispositivos.

**6. Ahora que comprende ARP, busque RARP y determine cuándo se podría utilizar ese protocolo.**

El RARP es utilizado para solicitar la dirección IP de una máquina física en una LAN.

**7. ¿A través de qué dispositivos de hardware se comunican las computadoras de los estudiantes con las computadoras de la administración?**

Los estudiantes se comunican a través de dispositivos como conmutadores (Switches), concentradores (Hubs), y enrutadores (Routers).

8. Si la computadora de un estudiante quiere enviar información a una computadora de administración, ¿qué información necesita para comenzar? ¿Cuáles son los posibles procesos utilizados para encontrar la computadora de administración?

Se necesitaría la tabla de direcciones MAC, el cache ARP y las solicitudes ARP

9. ¿Qué podría suceder si un ingeniero de redes configura incorrectamente un dispositivo asignándole una dirección IP incorrecta o conectándolo al puerto incorrecto?

La conectividad fallaría, impidiendo la conexión local o a Internet, y sería necesario corregir los errores para restablecer la conexión.

10. ¿Qué configuración de red le gustaría automatizar usando SDN?

La seguridad de Red: Automatizar la aplicación de políticas de seguridad, la detección de amenazas y la respuesta a incidentes de seguridad en tiempo real.

11. ¿Qué características de REST hacen que su uso sea sencillo?

- a) **Interfaz HTTP:** REST puede ser utilizado en cualquier cliente HTTP1, lo que facilita su implementación en una amplia variedad de sistemas.
- b) **Uso de protocolos estándar:** REST se basa en el protocolo HTTP, uno de los protocolos más antiguos2. Esto permite el intercambio y la recuperación de información entre sistemas de manera estandarizada3.
- c) **Sin estado (Stateless):** Un servicio REST no tiene estado, lo que significa que entre dos llamadas cualesquiera, el servicio pierde todos sus datos4. Esto implica que el cliente debe mantener el estado y pasarlo en cada llamada, lo que puede parecer tedioso, pero ofrece una gran ventaja en términos de escalabilidad4.
- d) **Simplicidad:** Comparado con otras arquitecturas como XML-RPC o SOAP, REST es mucho más simple1. Emplea una interfaz web que usa hipertextos para la representación y transición de la información1, lo que simplifica el proceso de desarrollo.