



**Universidad Tecnológica
del Norte de Guanajuato**

Organismo Público Descentralizado del Gobierno del Estado

"Educación y progreso para la vida"

TÉCNICO SUPERIOR UNIVERSITARIO EN
TECNOLOGÍAS DE LA INFORMACIÓN ÁREA INFRAESTRUCTURA DE REDES
DIGITALES.

PROGRAMACIÓN DE REDES

UNIDAD III PROGRAMACIÓN DE REDES

PROFESOR: GABRIEL BARRÓN RODRÍGUEZ

LABORATORIO "ENTERPRISE NETWORKING"

NOMBRE DEL ALUMNO: JOSÉ MANUEL GUTIÉRREZ PÉREZ

GRUPO: GIRO541

- ✓ **¿Por qué algunos dispositivos funcionan en unas redes y otros no?**
 - para que un dispositivo funcione en una red, debe contar con una tarjeta NIC compatible y tener una dirección MAC única y una dirección IPv4 asignada correctamente.
- ✓ **¿Cómo se encuentran los dispositivos entre sí?**
 - se encuentra mediante solicitudes de mensajes ip ala puerta de predeterminada que pertenece al enrutador y el enrutador hace lo siguiente
- ✓ **¿Qué son conmutadores, enrutadores e interfaces y cómo se configuran?**
 - Un conmutado: Es un dispositivo de capa 2 (capa de enlace de datos), donde la mayoría de las interfaces no tienen direcciones IPv4, se configura mediante vlan donde se deben de crear.
 - Enrutadores: es un dispositivo de capa 3 (capa de red). La mayoría de sus interfaces tiene una dirección ipv4, se configura las interfaces y algunas otras funciones, como la tabla de enrutamiento y traducción de direcciones red
 - Interfaces: son todos los puertos físicos RJ45 donde se pueden configurar, para configurar extensos, pero admite creación de interfaces virtuales, puertos troncales, LAN virtuales, etc.
- ✓ **¿Cómo se relacionan REST y SDN con esta información?**
 - SDN utiliza un software de red para configurar la red donde agregar nuevas funciones y protocolos a los dispositivos de red, esta característica y protocolos permite que las aplicaciones de los terceros se puedan configuren dinámicamente los dispositivos y defina como lo dispositivos procesan los datos
 - REST esta diseñado para simplificar las llamadas a funciones muchas api nuevas se basa en el protocolo de transferencia de datos representacional(REST)
- ✓ **Si un Switch es más eficiente que un Hub, ¿por qué comprarías un Hub?**
 - el concentrador recibe los datos y los envía a cada dispositivo que está conectado a él. Si uno de los dispositivos tiene la dirección IP coincidente, responde.
- ✓ **f. Ahora que comprende ARP, busque RARP y determine cuándo se podría utilizar ese protocolo.:**
 - permite que un dispositivo como una computadora sin disco duro o impresora de red obtenga una dirección ip cuando solo se conoce la dirección MAC, en pocas palabras e utiliza para solicitar una dirección ip
- ✓ **g. ¿A través de qué dispositivos de hardware se comunican las computadoras de los estudiantes con las computadoras de la administración?**
 - Se pueden conectar mediante diferentes dispositivos pero los que se usan normal mente son los switches, hub y routers los cuales por los que se pueden comunicar los alum nos con el administrador
- ✓ **h. 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?

- La información que se necesita para comenzar es cuatro los cuales son los siguientes:
- Proceso uno: Tabla de direcciones MAC, Proceso dos: caché ARP Proceso tres: Solicitud ARP, Proceso cuatro: tabla de direcciones MAC
- ✓ **¿Qué podría pasar si un ingeniero de redes configura mal un dispositivo asignándole una dirección IP incorrecta o conectándolo al puerto incorrecto?**
 - No tendría conectividad ya que no tendría conexión internet y por lo tanto no tendría acceso al internet ni con las demás redes conectadas
- ✓ **j. ¿Qué configuración de red le gustaría automatizar usando SDN?**
 - La optimización de rutas ya Puedes ajustar automáticamente las rutas de red en función de métricas como la latencia, el ancho de banda y otros parámetros de rendimiento.
- ✓ **k. ¿Qué características de REST hacen que su uso sea sencillo?**
 - Interfaz Uniforme: REST sigue una interfaz uniforme que simplifica las llamadas a funciones. Esta uniformidad se traduce en operaciones estándar de HTTP (GET, POST, PUT, DELETE) que son fáciles de entender y utilizar.
 - Independencia de Plataforma: REST permite la independencia de plataforma, lo que significa que los servicios web desarrollados en REST pueden ser consumidos por diferentes tipos de clientes, como aplicaciones web, móviles o incluso servicios de terceros, sin depender de una tecnología específica.
 - Sin Estado (Stateless): La arquitectura REST es sin estado, lo que significa que cada solicitud del cliente al servidor contiene toda la información necesaria para entender y procesar la solicitud. Esto simplifica el manejo del estado del sistema en el servidor y mejora la escalabilidad.