

**Examen parcial 2**

**Álvaro Martín Romero**

# Índice

[Misión 1: Reconexión en la Base Eco (Hoth) – Direccionamiento IP y Subredes](#_Misión_1:_Reconexión)

[Misión 2: Sabiduría de Yoda – Algoritmos de Enrutamiento y Rutas](#_Misión_2:_Sabiduría)

[Misión 3: Los Nombres del Holonet – DNS y Resolución de Nombres](#_Misión_3:_Los)

[Misión 4: “Es una trampa… de protocolos!” – TCP vs UDP en las transmisiones](#_Misión_4:_“Es)

[Misión 5: Comunicación Segura o lado oscuro – Criptografía y Seguridad de la Red](#_Misión_5:_Comunicación)

## Misión 1: Reconexión en la Base Eco (Hoth) – *Direccionamiento IP y Subredes*

**Paso 1: Identificar necesidades de hosts por subred**

| **Departamento** | **Hosts requeridos** | **Tamaño mínimo de subred** | **Hosts útiles** |
| --- | --- | --- | --- |
| Comando Central | 50 | /26 | 62 |
| Defensa Perimetral | 30 | /27 | 30 |
| Centro Médico | 20 | /27 | 30 |
| Hangar y Taller | 14 | /28 | 14 |
| Enlace Troncal (antena) | 2 | /30 | 2 |

*Se ordenan de mayor a menor requerimiento para asignar las subredes más grandes primero.*

**Paso 2: Asignación de subredes**

**Red base:** 172.16.0.0/24

**Rango total:** 172.16.0.0 - 172.16.0.255  
Direcciones disponibles: 256

🔹 1. Comando Central (~50 hosts)

* Subred: 172.16.0.0/26
* Rango de hosts: 172.16.0.1 – 172.16.0.62
* Broadcast: 172.16.0.63
* Hosts útiles: 62

🔹 2. Defensa Perimetral (~30 hosts)

* Subred: 172.16.0.64/27
* Rango de hosts: 172.16.0.65 – 172.16.0.94
* Broadcast: 172.16.0.95
* Hosts útiles: 30

🔹 3. Centro Médico (~20 hosts)

* Subred: 172.16.0.96/27
* Rango de hosts: 172.16.0.97 – 172.16.0.126
* Broadcast: 172.16.0.127
* Hosts útiles: 30

🔹 4. Hangar y Taller (~14 hosts)

* Subred: 172.16.0.128/28
* Rango de hosts: 172.16.0.129 – 172.16.0.142
* Broadcast: 172.16.0.143
* Hosts útiles: 14

🔹 5. Enlace Troncal Interplanetario (2 hosts)

* Subred: 172.16.0.144/30
* Rango de hosts: 172.16.0.145 – 172.16.0.146
* Broadcast: 172.16.0.147
* Hosts útiles: 2

**Resumen Final del Plan de Direccionamiento**

| **Subred** | **Dirección de red** | **Máscara CIDR** | **Rango de hosts** | **Broadcast** | **Hosts útiles** |
| --- | --- | --- | --- | --- | --- |
| Comando Central | 172.16.0.0 | /26 | 172.16.0.1 – 172.16.0.62 | 172.16.0.63 | 62 |
| Defensa Perimetral | 172.16.0.64 | /27 | 172.16.0.65 – 172.16.0.94 | 172.16.0.95 | 30 |
| Centro Médico | 172.16.0.96 | /27 | 172.16.0.97 – 172.16.0.126 | 172.16.0.127 | 30 |
| Hangar y Taller | 172.16.0.128 | /28 | 172.16.0.129 – 172.16.0.142 | 172.16.0.143 | 14 |
| Enlace Troncal (antena) | 172.16.0.144 | /30 | 172.16.0.145 – 172.16.0.146 | 172.16.0.147 | 2 |

**Observaciones**

* Aún quedan direcciones sin usar desde 172.16.0.148 a 172.16.0.255 para futuras expansiones o enlaces secundarios.
* Cada subred fue ajustada con precisión para minimizar desperdicio de IPs.
* El enlace troncal /30 es perfecto para conectar dos interfaces de routers.

## Misión 2: Sabiduría de Yoda – Algoritmos de Enrutamiento y Rutas

**Diferencias entre Enrutamiento Estático y Dinámico**

| **Característica** | Enrutamiento Estático | Enrutamiento Dinámico |
| --- | --- | --- |
| **Configuración** | Manual: el administrador define todas las rutas a mano | Automática: los routers intercambian información entre ellos |
| **Adaptabilidad** | No se adapta a cambios automáticamente | Se ajusta ante cambios en la topología |
| **Escalabilidad** | Difícil de escalar en redes grandes o en expansión galáctica | Adecuado para redes medianas y grandes |
| **Recurso del sistema** | Requiere pocos recursos del router | Consume CPU, memoria y ancho de banda |
| **Seguridad y control** | Mayor control, menor riesgo de rutas erróneas | Menor control directo, aunque más flexible |
| **Ejemplo de uso** | Rutas de respaldo, enlaces simples o específicos | Redes empresariales, campus, infraestructura crítica |
| **Ejemplo de reacción ante caída** | No responde: ruta se mantiene aunque el destino esté inactivo | Recalcula rutas alternativas automáticamente |

**Enrutamiento Dinámico: Protocolo Ejemplo – OSPF**

**OSPF (Open Shortest Path First) es un protocolo dinámico de estado de enlace que:**

* Calcula la ruta más corta con el algoritmo de Dijkstra.
* Divide la red en áreas (como sectores galácticos).
* Es más rápido y eficiente que protocolos más antiguos como RIP.
* Detecta cambios y actualiza solo lo necesario, sin reenviar toda la tabla.

**Vector de Distancia vs Estado de Enlace**

| **Enfoque** | Vector de Distancia (ej: RIP) | Estado de Enlace (ej: OSPF) |
| --- | --- | --- |
| **Funcionamiento** | Cada router dice a sus vecinos qué rutas conoce y a qué distancia | Cada router comparte el "mapa" completo de su entorno local |
| **Convergencia** | Lenta (puede tardar en adaptarse a cambios) | Rápida y precisa |
| **Consumo de red** | Menor, pero menos eficiente en redes grandes | Mayor al inicio, pero más optimizado a largo plazo |
| **Precisión de rutas** | Menor: solo conoce "lo que le dicen" | Mayor: calcula rutas óptimas con toda la información de la red |
| **Escalabilidad** | Limitada (máx. 15 saltos en RIP) | Alta (ideal para redes grandes como la HoloRed rebelde) |

**¿Cuál elegir?**

* **Usa enrutamiento estático si:**
  + La red es pequeña o no cambia frecuentemente.
  + Necesitas control absoluto.
  + Hay enlaces dedicados y específicos, como hacia una base secreta rebelde.
* **Usa enrutamiento dinámico si:**
  + La red es grande o sufre cambios regulares.
  + Necesitas resiliencia y adaptación.
  + La flota se mueve de planeta en planeta y las rutas deben actualizarse automáticamente.

## Misión 3: Los **Nombres del Holonet** – DNS y Resolución de Nombres

**¿Qué es el Sistema de Nombres de Dominio (DNS)?**

El **DNS (Domain Name System)** es como la *agenda galáctica de contactos* de la red. Traduce **nombres legibles por humanos** (como holonet.rebelion.org) en **direcciones IP** (como 172.16.23.5) que las computadoras pueden entender y utilizar para comunicarse.

Sin DNS, deberías recordar y teclear direcciones IP para cada planeta, estación o nodo. Como intentar llamar a un droide por su número de serie en lugar de su nombre.

**¿Cómo funciona la resolución de nombres?**

Cuando un terminal rebelde intenta contactar a holonet.rebelion.org, sigue este proceso:

**1. Consulta del Cliente**

El dispositivo (ej. una consola en Home One) pregunta:  
*"¿Cuál es la IP de holonet.rebelion.org?"*

**2. Revisión local (caché/DNS local)**

Primero revisa si ya conoce la IP (en su caché). Si no...

**3. Petición al Servidor DNS**

El dispositivo contacta a un **servidor DNS** (como un droide traductor de nombres) configurado en su red (a menudo un servidor local rebelde o uno externo confiable).

**4. Resolución jerárquica**

Si ese servidor no tiene la respuesta, consulta a otros servidores en esta secuencia:

* **Root servers** (nivel galáctico)
* **Servidores TLD** (.org, .net, etc.)
* **Servidores autoritativos** (quienes tienen los registros finales de rebelion.org)

**5. Respuesta y uso**

El servidor responde con la IP asociada, por ejemplo:  
*"holonet.rebelion.org = 172.16.23.5"*

Esa IP es usada para establecer conexión.

**¿Qué es un registro DNS?**

Un **registro DNS** es una entrada específica que relaciona un nombre con información. Algunos tipos:

| **Tipo de Registro** | **Función** | **Ejemplo** |
| --- | --- | --- |
| **A** | Traduce nombre a IP IPv4 | holonet.rebelion.org → 172.16.23.5 |
| **AAAA** | Traduce nombre a IP IPv6 | base.echo → 2001:db8::1234 |
| **CNAME** | Alias de otro nombre | status.rebelion.org → holonet.rebelion.org |
| **MX** | Dirección del servidor de correo | mail.rebelion.org |

**¿Qué pasa si el servidor DNS no está disponible?**

Si el servidor DNS falla o está mal configurado:

* Los nombres **no pueden resolverse** → dispositivos no saben a qué IP conectarse.
* Servicios como web, correo o SSH por nombre **dejan de funcionar**.
* Aún puedes comunicarte usando IPs **directamente**, pero eso es poco práctico (¿recordarías todas las IPs rebeldes?).

Para la Alianza, perder el DNS es como perder la Holored: **sin él, la flota no puede coordinarse**.

**Ejemplo rebelde completo**

Nombre: holonet.rebelion.org

Registro A: 172.16.23.5

Cuando Home One intenta conectarse a holonet.rebelion.org, consulta al servidor DNS rebelde local, recibe la IP 172.16.23.5, y establece comunicación con el nodo de la HoloRed. Misión cumplida.

## Misión 4: *“Es una trampa… de protocolos!”* – TCP vs UDP en las transmisiones

**TCP vs UDP – Dos caminos de la Fuerza (de transmisión)**

| **Característica** | TCP (Transmission Control Protocol) | UDP (User Datagram Protocol) |
| --- | --- | --- |
| **Tipo de conexión** | Orientado a conexión | No orientado a conexión |
| **Fiabilidad** | Alta: garantiza entrega completa y en orden | Baja: no garantiza entrega ni orden |
| **Control de flujo** | Sí | No |
| **Control de congestión** | Sí | No |
| **Orden de los datos** | Asegurado | No garantizado |
| **Velocidad** | Menor (por los controles y confirmaciones) | Mayor (sin controles ni confirmaciones) |
| **Sobrecarga (overhead)** | Alta | Baja |
| **Uso típico** | Transferencia segura de datos importantes | Comunicación rápida y ligera |

**TCP: El Protocolo Jedi – Preciso y confiable**

**¿Por qué se considera confiable?**

* TCP establece una conexión (proceso *three-way handshake*).
* Usa números de secuencia para ordenar los datos.
* Exige confirmaciones (ACKs): si un paquete no llega, se retransmite.
* Controla la congestión y adapta la velocidad para no saturar la red.

**Ejemplos en la Galaxia:**

* Transferencia de los planos de la Estrella de la Muerte → deben llegar completos y sin errores.
* Correos rebeldes entre bases (SMTP/IMAP).
* Comando y control desde Home One (por ejemplo, configuración remota por SSH).
* Recuperación de datos en R2-D2 vía FTP o HTTP.

**UDP: El Protocolo Caza Estelar – Rápido y directo**

**¿Por qué es más rápido?**

* UDP no establece conexión previa: simplemente envía los datos.
* No hay confirmaciones, ni reintentos, ni verificación de orden.
* Esto reduce latencia, ideal cuando lo más importante es la velocidad.

**Ejemplos en la Galaxia:**

* Stream de vídeo desde una X-Wing → si se pierde un fotograma, no importa: el vídeo sigue.
* Coordenadas de combate en tiempo real entre cazas → mejor perder una que llegar tarde.
* Señales de navegación o radar activo (como sonar galáctico).
* Juegos en red simulados o entrenamiento rebelde (multijugador).

**¿Qué implican estas diferencias?**

| **Pregunta** | **TCP** | **UDP** |
| --- | --- | --- |
| ¿Qué pasa si se pierde un paquete? | Se retransmite automáticamente | Se pierde para siempre |
| ¿Y si llegan fuera de orden? | Se reordena | Se entregan como llegan (sin corrección) |
| ¿Y si hay congestión en la red? | Reduce velocidad para evitar más problemas | Sigue enviando igual (puede saturar más) |
| ¿Qué prefieres para una transmisión crítica? | Seguridad (TCP) | Velocidad (UDP) si la pérdida es tolerable |

## Misión 5: Comunicación Segura o lado oscuro – *Criptografía y Seguridad de la Red*

**Cifrado Simétrico vs Asimétrico – Dos filosofías del cifrado rebelde**

**1. Cifrado Simétrico – Una sola clave compartida**

**Ejemplo galáctico:** Leia y Luke comparten la frase clave “YodaSabeMás” para cifrar y descifrar sus holomensajes.

* Usa una misma clave secreta para cifrar y descifrar.
* Rápido y eficiente, ideal para grandes volúmenes de datos.
* **Problema:** la clave debe compartirse de forma segura, lo cual puede ser arriesgado si la red es vulnerable.

| **Ventajas** | **Desventajas** |
| --- | --- |
| Rápido y con poca carga de CPU | Requiere canal seguro para compartir la clave |
| Ideal para datos continuos | Si se filtra la clave, toda la comunicación se compromete |

Protocolos comunes: AES, DES, ChaCha20

**2. Cifrado Asimétrico – Par de claves: pública y privada**

**Ejemplo galáctico**: La base rebelde en Yavin IV quiere enviar un mensaje secreto a un nuevo aliado sin conocerlo antes. Usa la clave pública del aliado para cifrar el mensaje. Solo el aliado podrá descifrarlo con su clave privada.

* Se usa una clave pública para cifrar, y una clave privada (única, secreta) para descifrar.
* No es necesario intercambiar claves secretas antes.
* Más lento que el simétrico, pero ideal para establecer comunicaciones seguras iniciales.

| **Ventajas** | **Desventajas** |
| --- | --- |
| No necesita compartir secretos | Más lento y consume más recursos |
| Permite autenticación y firma | No ideal para cifrado de grandes volúmenes |

Protocolos comunes: RSA, ECC (criptografía de curva elíptica), Diffie-Hellman

**Autenticación y No Repudio – La Fuerza del mensaje verdadero**

**Autenticación**

* Permite verificar la identidad del emisor.
* Ejemplo: Luke firma digitalmente un mensaje → Leia puede estar segura de que lo envió él y no un Sith impostor.

**No repudio**

* El emisor no puede negar que envió el mensaje.
* Ejemplo: Si Mon Mothma firma digitalmente una orden de ataque, no puede decir después que no la envió.

Estas propiedades se logran con firmas digitales, basadas en criptografía asimétrica y hashes (como SHA-256).

**Seguridad en Administración Remota**

**Telnet**

* Envía datos en texto plano, incluyendo contraseñas.
* Cualquier espía puede interceptar y leer los datos.

**SSH (Secure Shell)**

* Usa cifrado fuerte (normalmente asimétrico para establecer conexión y luego simétrico para la sesión).
* Protege la autenticación, los comandos y la transferencia de archivos.
* Ideal para administrar routers, servidores o droides desde bases remotas.

**Ejemplo Final:** Comunicación Segura Rebelde

Supón que la Alianza quiere contactar con un nuevo planeta aliado (como Lothal):

1. El aliado publica su clave pública.
2. La Alianza cifra un mensaje con esa clave y lo envía.
3. Solo el destinatario puede descifrarlo con su clave privada.
4. La Alianza firma el mensaje con su clave privada → el aliado puede verificarla con la clave pública rebelde.
5. El mensaje es secreto, auténtico y confiable. ¡El Imperio no puede leerlo!