

## Control 3

## Redes

**Plazo de entrega: 20 de noviembre 2024**

*José M. Piquer*

### Introducción

Las preguntas deben responderse en lenguaje directo, técnico y evitar las frases que suenan bien pero no dicen nada.

Un ejemplo sacado de chatGPT es: “Es importante asegurarse de que el enrutamiento, la configuración de NAT y las políticas de seguridad estén correctamente implementadas para garantizar una comunicación fluida y segura tanto dentro como fuera de la red corporativa”.

Si se fijan, esa frase no dice nada concreto, nada que permita evaluar si Uds entienden la materia (sabemos que chatGPT NO la entiende).

Si Uds escriben algo así en el control, tendrán cero puntos por ello.

### P1: TCP avanzado

#### 1.1 Protocolo de Conexión y fin

El protocolo de conexión y desconexión de TCP es simple, pero tiene fragilidades que se han ido mejorando en el tiempo. Busque en Internet estos temas y responda las siguientes preguntas:

- Hubo un ataque famoso a TCP que se llamó Syn Flood. Explique cómo funcionaba y busque cómo se puede resolver, dejando fuera la solución de *cookies* que es la siguiente pregunta, y también dejando fuera las soluciones tipo firewall o proxies. Analice las alternativas que se pueden implementar en un servidor y elija la que le parece mejor, explicando por qué.
- Averigüe y explique cómo funcionan las *Syn Cookies* para resolver este ataque. ¿Cómo se compara esta solución con la que Ud escogió antes? Justifique cuál usaría.

- El número de secuencia inicial de una conexión TCP se recomienda que sea un número aleatorio. ¿Por qué no es buena idea partir siempre de cero?
- El cerrar una conexión es siempre difícil: si yo recibo un FIN y envío el ACK, ¿puedo olvidarme de esa conexión? ¿qué pasa si mi ACK se pierde? ¿cómo se resuelve en TCP?
- En TCP, inyectar un paquete falso en una conexión es casi imposible. En UDP es muy fácil (como vimos en la T3). Proponga una extensión que podríamos hacerle al protocolo de la T2 para que fuera resistente a inyecciones de este tipo. ¿Qué tan resistente sería comparado con TCP?

## 1.2 Control de Flujo y Congestión

Responda las siguientes preguntas:

- ¿Por qué no me basta con las ventanas de envío/recepción y necesito control de flujo?
- ¿Qué pasa si sólo achico la ventana de recepción en vez de usar control de flujo?
- El simulador del curso no tiene control de flujo, ¿por qué no lo necesita?
- Hay un mito urbano: si tu descarga se pone muy lenta en un momento, mejor máatala y reintenta todo de nuevo y será más rápida. ¿Puede ser cierto?
- Hay gente que ha medido que varias conexiones TCP en paralelo funcionan mejor que una solo (ej: separo un archivo en tres partes del mismo tamaño y activo tres conexiones en paralelo). ¿Puede ser cierto? Argumente por qué.

## 1.3 Manejo de Timeouts

Responda las siguientes preguntas:

- ¿Qué pasa si el valor de mi timeout es demasiado pequeño?
- ¿Qué pasa si es muy grande?

- Si hay que elegir: ¿es mejor errar y definir uno muy grande o uno muy chico?
- ¿Por qué TCP implementa el algoritmo de Karn?
- Se ha propuesto usar una extensión de TCP llamada *timestamps* para mejorar. Explique cómo funciona y argumente si es buena o mala idea.
- Una propuesta alternativa sería numerar las retransmisiones: 0 es el primer envío y luego se incrementa para cada re-transmisión. Proponga un esquema que use ese esquema para recalcular los timeouts correctamente siempre.

## P2: Redes IP

### 2.1 Reserva de Recursos

Un concepto clave de IP es siempre asignar dinámicamente los recursos a las conexiones, según sus demandas instantáneas. La idea antigua de la telefonía era todo lo contrario: la conexión reservaba el ancho de banda necesario a lo largo de toda la duración y a lo largo de todo el camino entre el origen y el destino.

En Internet, esto sería equivalente a que la conexión TCP reservara memoria y ancho de banda en todos los routers por los que pasa.

Muchos creen que el éxito de Internet frente a la red telefónica se debió justamente a esta diferencia.

Argumente por qué esto es tan importante, y discuta si puede haber sido esta diferencia la clave del éxito de Internet.

### 2.2 IPv6

Todavía hoy en día una empresa que se conecta a Internet pide una dirección IPv4 pública y válida, y ya es casi imposible de conseguir.

- ¿Por qué no basta con obtener una dirección IPv6 pública?
- ¿Qué se podría hacer para que una empresa use solamente direcciones IPv6 públicas en su red?
- La empresa duda entre usar direcciones IPv6 públicas o privadas en su red interna. ¿Qué recomendaría Ud? ¿Por qué?

## 2.3 Máscara de Red

Al comienzo de Internet, las direcciones IP tenían una clase que indicaba de qué largo era el prefijo de red. Luego vino CIDR y hubo que agregar máscaras de red a las tablas de rutas.

Un ingeniero propone ahora mejorar los paquetes IP haciendo que las direcciones de origen y destino lleven el largo del prefijo de red de cada una. Así, un paquete que va de la dirección 192.80.24.1 a la dirección 146.83.1.1, ambas redes con máscara 255.255.255.0 iría como: origen: 192.80.24.1/24, destino: 146.83.1.1/24.

Obviamente esto tiene un costo de algunos bits en el header. Responda las siguientes preguntas:

- ¿En qué ayudaría saber cuántos bits de la dirección son de red y cuántos de host?
- En los routers de Internet, ¿podría ser más eficiente la búsqueda de rutas?
- ¿Podríamos eliminar los números de bits en los prefijos que manejan los routers en sus tablas, ya que vienen en las direcciones?
- ¿Hay algo que se puede echar a perder al implementar algo así?
- ¿Sería buena idea, al final?

## P3: Ruteo

### 3.1 Tablas de Rutas

Una regla importante en el ruteo es que siempre se busca primero la ruta más específica.

Responda las siguientes preguntas:

- ¿Por qué es así?
- Suponga que un router tiene una tabla de rutas ordenada, con las rutas más específicas primero y las menos la final. Escriba el pseudo-código de una función que busca una dirección IP en la tabla para encontrar la ruta hacia ella:

```
route = find_route(IP_dest, route_table)
```

### 3.2 BGP-4

Un ingeniero propone la idea de que todos los routers que corren BGP-4 envíen la información que tienen sobre los vectores de caminos de los sistemas autónomos que poseen a un sitio central.

De esa forma, ese sitio central podría aplicar el algoritmo de Dijkstra (como OSPF) para calcular los caminos óptimos en Internet. Como sólo considera Sistemas Autónomos, no sería tan gigante como el grafo de todos los prefijos de red.

Discuta esta idea y argumente técnicamente.