Redes de Computadores II

Curso 14/15 :: ordinario

Escuela Superior de Informática

2023/03/07 10:52:03

Este examen consta de 15 preguntas con un total de 35 puntos. Tres preguntas de test erróneas restan un punto. Sólo una opción es correcta a menos que se indique algo distinto. No está permitido el uso de calculadora.

Apellidos: SOLUCIÓN Nombre: _____ Grupo: _____

1. [1p] ¿Por qué NAT no tiene sentido en una red IPv6?
   - [x] **a**) NAT se creó principalmente para compensar la escasez de direcciones de IPv4.
   - [ ] **b**) Los encaminadores IPv6 no podrían manejar tablas NAT tan grandes.
   - [ ] **c**) No se pueden traducir las direcciones IPv6 puesto que las direcciones públicas y privadas tienen tamaños distintos.
   - [ ] **d**) NAT tiene sentido y se utiliza masivamente en IPv6.

2. [1p] A diferencia de IPv4, en IPv6 los dispositivos pueden autoconfigurarse con una dirección lógica local al enlace con plenas garantías de unicidad. ¿Cuál es el motivo?
   - [ ] **a**) Las direcciones IPv4 no pueden encapsularse en direcciones IPv6.
   - [ ] **b**) Las direcciones de red IPv4 requieren una dirección de broadcast.
   - [ ] **c**) Las direcciones IPv6 tienen una semilla aleatoria con una probabilidad de colisión inferior al 1 por millón.
   - [x] **d**) Las direcciones IPv4 tienen una longitud menor a las direcciones MAC habituales.

3. [1p] En igualdad de condiciones, un encaminador IPv6 es más rápido que uno IPv4 ¿por qué?
   - [ ] **a**) No fragmenta los datagramas reenviados
   - [ ] **b**) No calcula el checksum para cada paquete que reenvía
   - [x] **c**) No tiene que fragmentar ni calcular checksums
   - [ ] **d**) No es correcto. Los encaminadores IPv6 son más lentos porque manejan direcciones más grandes.

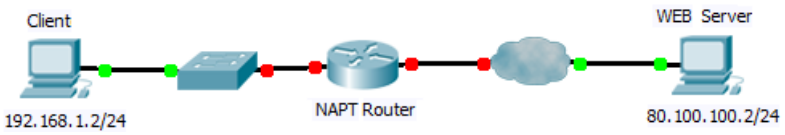
4. [1p] ¿Cuál es el propósito de la tabla NAT?
   - [ ] **a**) Proporcionar un servicio de directorio compatible con mi configuración LAN estándar.
   - [x] **b**) Determinar qué host de la red privada es el destinatario del paquete que recibe el encaminador.
   - [ ] **c**) Establecer la correspondencia entre las direcciones lógicas y físicas en la red privada.
   - [ ] **d**) Almacenar temporalmente las correspondencias IP a nombre de dominio consultadas al DNS.

5. [2p] Explica similitudes y diferencias entre una red privada y una VPN.

   Ambas tratan de proporcionar conectividad IP entre los computadores de una organización. La red privada utiliza infraestructura exclusiva de la organización (alquilada o propia) mientras que la VPN utiliza túneles a través de una red pública (típicamente Internet).

6. [1p] En una conexión TCP, en un determinado instante, se han recibido 3 segmentos con el mismo valor de ACK y ventana de recepción igual a 2 MSS. ¿Cuál es el estado de la red?
   - [ ] **a**) No hay ningún fenómeno indeseable.
   - [x] **b**) Hay congestión, se va a iniciar la fase de evitación de la congestión.
   - [ ] **c**) Hay saturación en el receptor, se controla el flujo.
   - [ ] **d**) Hay congestión, se va a iniciar la fase de arranque lento.

7. [5p] Dada la topología de la figura, explique los cambios que sufren las cabeceras TCP e IP de una petición HTTP emitida por el host «Client» hacia el servidor web y su correspondiente respuesta, teniendo en cuenta la existencia del encaminador NAPT.



Las cabecera TCP e IP de los mensajes procedentes de «Client» con destino a «Web Server» contiene los siguientes datos:

- IP origen: 192.168.1.2; IP destino: 80.100.100.2
- puerto origen: 22000 (aleatorio); puerto destino 80

Al atravesar el encaminador, éste cambia la IP origen a la dirección externa del router, por ejemplo: 120.10.10.1 y probablemente también el puerto origen (por ejemplo: 43001).

La respuesta del servidor web por tanto irá dirigida a 120.10.10.1:43001. Consultando la tabla, el encaminador substituye esos datos por los valores de partida que aparecían en la petición y lo envía a «Client».

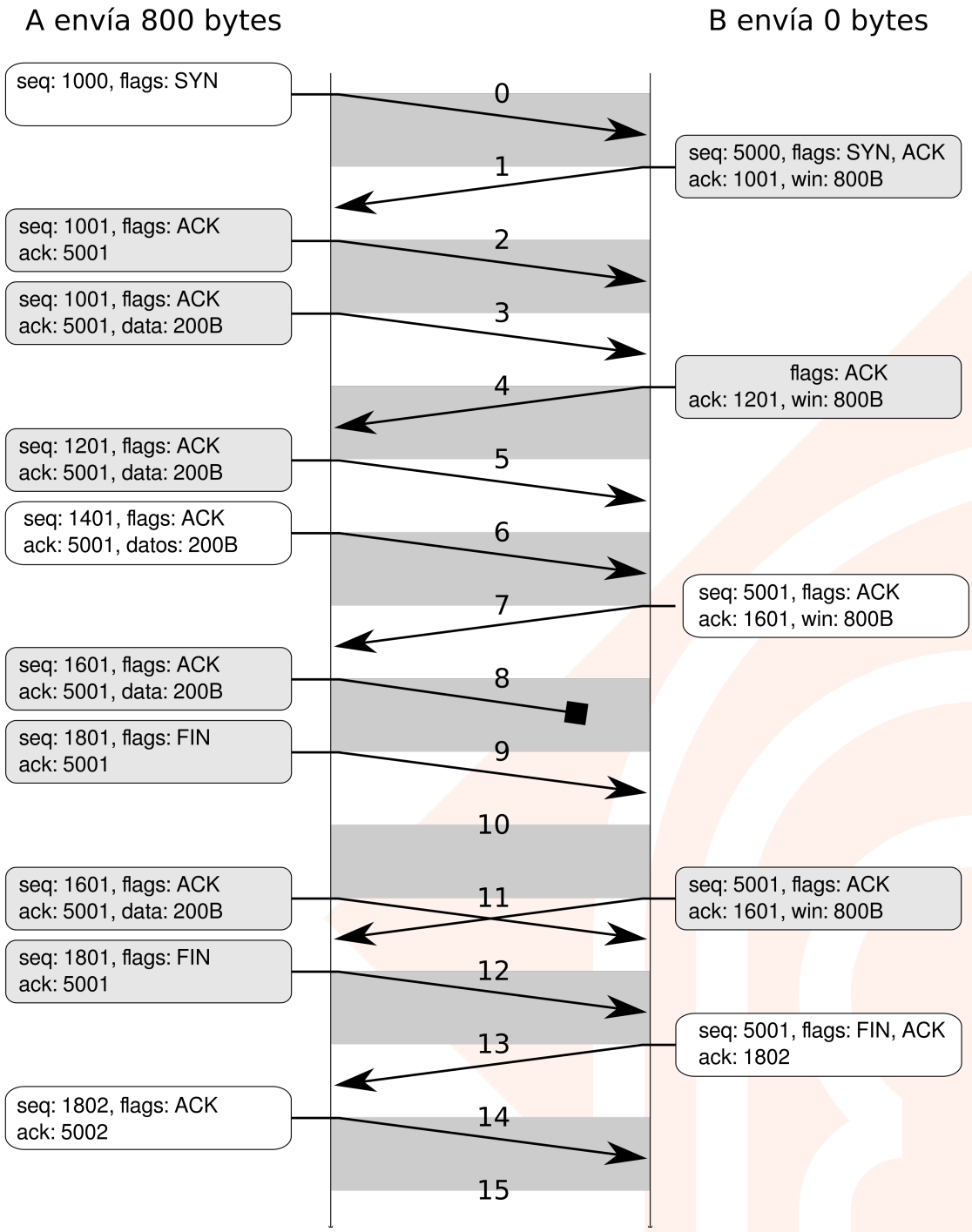
8. [4p] Explique con detalle cómo resuelve TCP los problemas de control de congestión y control de flujo, qué campos de la cabecera están implicados y la relación que tienen la ventana de recepción (rwnd) y la de congestión (cwnd).

TCP utiliza un sistema de ventana deslizante que numera los bytes de la carga útil del segmento. El número de secuencia del primer byte aparece en el campo «sequence number» de la cabecera. En un momento dado, el emisor solo puede enviar los bytes del buffer de salida que correspondan a los números de secuencia de la ventana de envío. Cuando un extremo de la conexión TCP recibe un segmento que lleva activo el flag ACK, el campo «acknowledgement» indica el número de secuencia del siguiente byte que espera recibir y por tanto la ventana de envío se puede desplazar y podrá enviar datos nuevos.

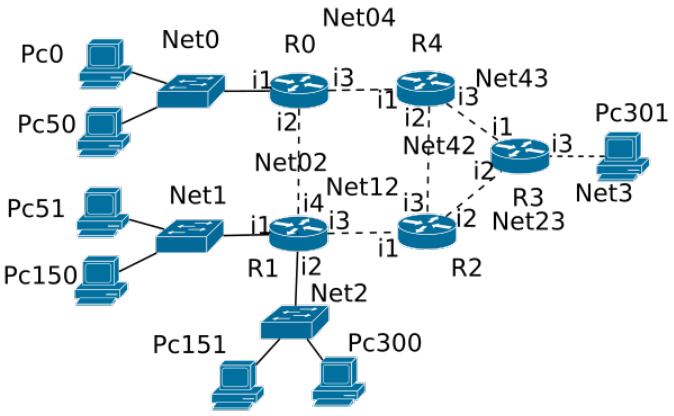
El tamaño de la ventana de envío se calcula como el mínimo(rwnd, cwnd), siendo *rwnd* la ventana de recepción y *cwnd* la ventana de envío. La ventana de recepción la controla el receptor mediante el campo «window» de la cabecera TCP y de ese modo implementa el mecanismo de control de flujo.

La ventana de congestión se recalcula continuamente y trata de adaptarse a los cambios que suceden en la red. Cuando se detecta congestión, la ventana de congestión se reduce y crece de nuevo con una tasa que depende de la gravedad del problema: mecanismo de arranque lento para timeout expirados o evitación de congestión en el caso de ACK duplicados. Éste es el mecanismo de control de congestión.

9. [5p] En la figura aparece un flujo TCP, incluyendo conexión y desconexión. Complete el contenido de los segmentos en blanco teniendo en cuenta que:
   - A está utilizando control de congestión.
   - B enviará un segmento a A cuando haya recibido dos segmentos de A desde el último segmento asentido o en el segundo tic de reloj desde el último segmento recibido.
   - El plazo de retransmisión de segmentos en A (timeout) es de 3 tics de reloj.
   - A usa un tamaño fijo de datos de 200 bytes.
   - B siempre enviará un valor de 800 en el campo de tamaño de la ventana de flujo.
   - A enviará segmentos con datos siempre que pueda.



10. [3p] En una red hay dispuestos 302 ordenadores (Pc0 a Pc301) y 5 routers (R0 a R4) con los interfaces i1 a i4 según la figura adjunta. Se proporciona la red 100.100.0.0/16 para direccionar por VLSM y minimizando el tamaño de las subredes.



En cada red, la primera dirección IP disponible se asignará al gateway y después a los PC empezando por el de identificador más bajo.

¿Cuáles son las direcciones de red (con máscara) de cada una de las subredes? Aparecen listadas de mayor a menor tamaño.

```
Net2   100.100.0.0/24
Net1   100.100.1.0/25
Net0   100.100.1.128/26
Net04  100.100.1.192/30
Net02  100.100.1.196/30
Net12  100.100.1.200/30
Net42  100.100.1.204/30
Net43  100.100.1.208/30
Net23  100.100.1.212/30
Net3   100.100.1.216/30
```

```
Net2   100.100.1.0/25
Net1   100.100.1.128/26
Net0   100.100.1.192/27
Net04  100.100.1.196/30
Net02  100.100.1.200/30
Net12  100.100.1.204/30
Net42  100.100.1.208/30
Net43  100.100.1.212/30
Net23  100.100.1.216/30
Net3   100.100.1.220/30
```

```
Net2   100.100.0.0/24
Net1   100.100.1.0/25
Net0   100.100.1.128/26
Net3   100.100.1.64/30
```

```
Net1   100.100.0.0/26
Net2   100.100.1.1/24
Net0   100.100.1.128/25
Net04  100.100.1.192/30
Net02  100.100.1.196/30
Net12  100.100.1.200/30
Net42  100.100.1.204/30
Net43  100.100.1.208/30
Net23  100.100.1.212/30
Net3   100.100.1.216/30
```

11. [2p] Sobre el ejercicio anterior, indique dirección IP, máscara y gateway de Pc151:

- ☐ **a)** 100.100.0.0 - 255.255.255.128 - GW: 100.100.1.1
- ☐ **b)** 100.100.0.2 - 255.255.255.128 - GW: 100.100.0.3
- ■ **c)** 100.100.0.2 - 255.255.255.0 - GW: 100.100.0.1
- ☐ **d)** 100.100.0.2 - 255.255.255.255 - GW: 100.100.2.1

12. [3p] Sobre la topología anterior, indique el contenido de la tabla de rutas (estática) de R1, donde IP(R2i1) es la IP de la interfaz i1 del router R2 y NetX representa la dirección de red y máscara de la red de dicho nombre.

```
dst       next-hop    iface
Net1      0.0.0.0     i1
Net2      0.0.0.0     i2
Net0      IP(R0i2)    i4
Net3      IP(R2i1)    i3
```

```
dst       next-hop    iface
Net1      0.0.0.0     i1
Net2      0.0.0.0     i2
Net02     0.0.0.0     i4
Net12     0.0.0.0     i3
Net0      IP(R0i2)    i4
Net3      IP(R2i1)    i3
Net04     IP(R0i2)    i4
Net42     IP(R2i1)    i3
Net43     IP(R2i1)    i3
Net23     IP(R2i1)    i3
```

```
dst       next-hop    iface
Net1      0.0.0.0     i1
Net2      0.0.0.0     i2
Net02     0.0.0.0     i3
Net12     0.0.0.0     i4
Net0      IP(R0i2)    i4
Net3      IP(R3i2)    i3
Net04     IP(R0i2)    i4
Net42     IP(R2i1)    i3
Net43     IP(R3i2)    i3
Net23     IP(R2i1)    i3
```

```
dst       next-hop    iface
Net1      0.0.0.0     i1
Net2      0.0.0.0     i2
Net02     0.0.0.0     i3
Net12     0.0.0.0     i4
Net0      IP(R1i4)    i4
Net3      IP(R1i3)    i3
Net04     IP(R1i4)    i4
Net42     IP(R1i3)    i3
Net43     IP(R1i3)    i3
Net23     IP(R1i3)    i3
```

13. [2p] A esta misma red se le aplica un protocolo de vector distancia con la métrica:
    - 0: para directamente conectados.
    - número de saltos: para el resto.

    Indique el vector distancia inicial de R1:

```
dst       next-hop   iface   metric
Net1      0.0.0.0    i1      0
Net2      0.0.0.0    i2      0
Net02     0.0.0.0    i3      0
Net12     0.0.0.0    i4      0
Net0      IP(R0i2)   i4      0
Net3      IP(R2i1)   i3      0
Net04     IP(R0i2)   i4      0
Net42     IP(R2i1)   i3      0
Net43     IP(R2i1)   i3      0
Net23     IP(R2i1)   i3      0
```

```
dst       next-hop   iface   metric
Net1      0.0.0.0    i1      0
Net2      0.0.0.0    i2      0
Net0      IP(R0i2)   i4      0
Net3      IP(R2i1)   i3      0
```

```
dst       next-hop   iface   metric
Net1      0.0.0.0    i1      0
Net2      0.0.0.0    i2      0
Net02     0.0.0.0    i4      0
Net12     0.0.0.0    i3      0
```

```
dst       next-hop   iface   metric
Net1      0.0.0.0    i1      0
Net2      0.0.0.0    i2      0
Net02     0.0.0.0    i3      0
Net12     0.0.0.0    i4      0
Net0      IP(R1i4)   i4      0
Net3      IP(R1i3)   i3      0
Net04     IP(R1i4)   i4      0
Net42     IP(R1i3)   i3      0
Net43     IP(R1i3)   i3      0
Net23     IP(R1i3)   i3      0
```

14. [2p] Sobre la misma red, indique el vector distancia inicial de R1 tras recibir el primer vector distancia de R2 (misma métrica que el ejercicio anterior).

dst	next-hop	iface	metric
Net1	0.0.0.0	i1	0
Net2	0.0.0.0	i2	0
Net02	0.0.0.0	i4	0
Net12	0.0.0.0	i3	0
Net23	IP(R2i1)	i3	1
Net42	IP(R2i1)	i3	1

dst	next-hop	iface	metric
Net1	0.0.0.0	i1	0
Net2	0.0.0.0	i2	0
Net0	IP(R0i2)	i4	0
Net3	IP(R2i1)	i3	0
Net23	IP(R2i1)	i4	1
Net42	IP(R2i1)	i4	1

dst	next-hop	iface	metric
Net1	0.0.0.0	i1	0
Net2	0.0.0.0	i2	0
Net02	0.0.0.0	i3	0
Net12	0.0.0.0	i4	0
Net0	IP(R0i2)	i4	0
Net3	IP(R2i1)	i3	0
Net04	IP(R0i2)	i4	0
Net42	IP(R2i1)	i3	0
Net43	IP(R2i1)	i3	0
Net23	IP(R2i1)	i3	0
Net23	IP(R2i1)	i4	1
Net42	IP(R2i1)	i4	1

dst	next-hop	iface	metric
Net1	0.0.0.0	i1	0
Net2	0.0.0.0	i2	0
Net02	0.0.0.0	i4	0
Net12	0.0.0.0	i3	0
Net0	IP(R0i2)	i4	1
Net3	IP(R2i1)	i3	2
Net04	IP(R0i2)	i4	1
Net42	IP(R2i1)	i3	1
Net43	IP(R2i1)	i3	2
Net23	IP(R2i1)	i3	1

15. [2p] Indique el vector distancia de R1 al final de la convergencia:

```
dst       next-hop iface    metric
Net1      0.0.0.0   i1      0
Net2      0.0.0.0   i2      0
Net02     0.0.0.0   i4      0
Net12     0.0.0.0   i3      0
Net0      IP(R0i2)  i4      1
Net3      IP(R2i1)  i3      2
Net04     IP(R0i2)  i4      1
Net42     IP(R2i1)  i3      1
Net43     IP(R2i1)  i3      2
Net23     IP(R2i1)  i3      1
```

```
dst       next-hop  iface   metric
Net1      0.0.0.0   i1      0
Net2      0.0.0.0   i2      0
Net0      IP(R0i2)  i4      1
Net3      IP(R2i1)  i3      2
```

```
dst       next-hop  iface   metric
Net1      0.0.0.0   i1      0
Net2      0.0.0.0   i2      0
Net02     0.0.0.0   i3      0
Net12     0.0.0.0   i4      1
Net0      IP(R0i2)  i4      1
Net3      IP(R3i1)  i2      2
Net04     IP(R0i2)  i4      1
Net42     IP(R2i1)  i3      1
Net43     IP(R2i1)  i3      1
Net23     IP(R2i1)  i3      2
```

```
dst       next-hop  iface   metric
Net1      0.0.0.0   i1      0
Net2      0.0.0.0   i2      0
Net02     0.0.0.0   i3      0
Net12     0.0.0.0   i4      0
Net0      IP(R1i4)  i4      1
Net3      IP(R1i3)  i3      1
Net04     IP(R1i4)  i4      1
Net42     IP(R1i3)  i3      1
Net43     IP(R1i3)  i3      2
Net23     IP(R1i3)  i3      1
```