

**Primer Parcial – 26 de setiembre de 2024**  
(ref: solucion\_prc20240926.odt)

**Instrucciones**

- Indique su nombre completo y número de cédula en cada hoja.
- Numere todas las hojas e indique en la primera la cantidad total de hojas que entrega.
- Escriba las hojas de un solo lado y utilice una caligrafía claramente legible.
- Comience cada pregunta en una hoja nueva.
- Sólo se responderán dudas de letra. No se responderán dudas de ningún tipo durante los últimos 15 minutos de la prueba.
- La prueba es individual y sin material. Apague su teléfono celular mientras esté en el salón de la prueba.
- Duración: 2 horas. Culminadas las 2 horas, el alumno no podrá modificar de ninguna forma las hojas.
- Justifique todas sus respuestas.

**Pregunta 1 (12 puntos)**

Suponga que desde la dirección *jefe@redes.uy* se envía un correo electrónico a *empleado@cables.uy*, con asunto “Deseo comprar cables” y cuerpo del mensaje “¿Cuánto salen 15 metros de cable Ethernet Cat. 6?”.

- a) ¿Cómo se puede obtener el servidor de correo asociado a *cables.uy*? Indique el protocolo y los mensajes intercambiados entre el *host* del usuario y el servidor que le proporcionará dicha información.
- b) Una vez obtenido el servidor de correo asociado a *cables.uy*, ¿qué protocolo se utiliza para enviar el correo electrónico mencionado? Indique la secuencia de comandos necesarios para realizar el envío desde el punto de vista del servidor de correos de *redes.uy*.
- c) Si *empleado@cables.uy* desea leer los correos electrónicos de su bandeja de entrada, ¿qué protocolos puede usar? Realice una breve explicación de cada uno y mencione qué diferencias hay entre ellos.

**Solución**

a) Con el protocolo DNS (Domain Name System), se hace una consulta al registro MX de *cables.uy*, y la respuesta será el hostname del servidor de correos de *cables.uy*. Luego se debe hacer una consulta al registro A de dicho servidor de correos para obtener su dirección IP y poder iniciar la comunicación.

b) Se utiliza SMTP (Simple Mail Transfer Protocol), un protocolo *push-only*, es decir que permite enviar correos pero no leerlos.

Para el mensaje, se debe seguir el formato establecido en RFC 822, con una cabecera y un cuerpo separados por una línea en blanco. La cabecera debe tener necesariamente los campos To, From y Subject.

Por lo tanto, el contenido del mensaje sería:

To: empleado@cables.uy  
From: jefe@redes.uy  
Subject: Deseo comprar cables

¿Cuánto salen 15 metros de cable Ethernet Cat. 6?”

La secuencia de comandos sería la siguiente, R denota al servidor SMTP de *redes.uy* (que en esta ocasión actúa como cliente) y C al servidor SMTP de *cables.uy*:

```
C 220 SMTP Server ready
R HELO redes.uy
C 250 cables.uy, pleased to meet you
R MAIL FROM: <jefe@redes.uy>
```

```
C 250 Sender OK
R RCPT TO: <empleado@cables.uy>
C 250 Recipient OK
R DATA
C 354 Start input
R <contenido del mensaje, mencionado anteriormente>
.
C 250 Message queued for delivery
R QUIT
C 221 Goodbye, hope to see you again
```

El final del mensaje en SMTP se marca con <CRLF>.<CRLF>, como se observa anteriormente.

c) Puede utilizar tanto IMAP (Internet Message Access Protocol) o POP (Post Office Protocol), dos protocolos de acceso a correo electrónico. POP es un protocolo *stateless*, en el sentido que no se almacena estado (como lectura de correos electrónicos) entre sesiones. Tiene dos modos de operación, *download and keep* y *download and delete*. En el primero los mensajes se obtienen del servidor de correo pero no se eliminan, y en el segundo se obtienen los mensajes y luego se eliminan del servidor de correo.

IMAP introduce la noción de carpetas asignadas a los correos electrónicos. Al llegar un correo al servidor, automáticamente se asigna a la carpeta por defecto INBOX, y luego se puede mover a otra carpeta creada por el usuario, como puede ser Leídos, Importantes, Facultad, Trabajo, etc.

### **Pregunta 2 (8 puntos)**

- Suponga que un navegador web obtiene un objeto mediante el protocolo HTTP. ¿Cómo sabe el tipo de este objeto para representarlo correctamente en pantalla? Por ejemplo, cómo sabe si es una imagen jpeg, gif, o un texto?
- Explique el mecanismo GET condicional comentando qué información se incluye para ello en la cabecera de HTTP, cómo es utilizada y cuál es el beneficio que se busca con su uso.

### **Solución**

- El campo Content-type del cabezal de la respuesta conteniendo el objeto indica su "MIME type", un estándar para describir tipos de archivos. Ejemplo: Content-type: image/jpeg
- Se trata de un mecanismo que permite a la caché verificar que sus objetos están actualizados. De esta manera, el beneficio perseguido es: bajar el ancho de banda consumido y mejorar los tiempos de respuesta.

Para ello se incluye en la cabecera una línea "If-Modified-Since:".

La caché almacena, junto con el objeto la fecha (y hora) incluida en la línea "Last-Modified:" de la respuesta.

Frente a una siguiente consulta por dicho objeto, la caché envía un GET condicional o sea, agregando al Get la línea "If-modified-since: fecha\_(y hora)\_recibida\_en\_Last-Modified:".

El servidor http que recibe la petición sólo devolverá el objeto en cuestión si fue modificado con posterioridad a dicha fecha (y hora).

### **Pregunta 3 (10 puntos)**

- Enumere y explique dos ventajas y dos desventajas que percibe una aplicación al usar UDP como protocolo de transporte, en comparación con utilizar TCP.
- Mencione una aplicación que utilice UDP como capa de transporte y a partir de su respuesta de la parte a), fundamente porque se podría haber tomado esa decisión de diseño.

### **Solución**

a)Ventajas:

\* La aplicación tendría mayor control sobre los datos que se envían y cuando lo hacen dado que UDP simplemente empaqueta los datos y los envía por la red sin ningún proceso intermedio como la transferencia fiable o el control de congestión que pueden hacer que los datos se retrasen en el

emisor antes de ser enviados. Por lo tanto, una ventaja concreta podría ser que el retardo sería menor.

\* No hay retardo en el establecimiento de conexión. Dado que UDP no realiza ningún establecimiento de conexión, no hay ningún proceso previo al envío de los datos que pueda agregar retardos.

\* Poca sobrecarga del host y overhead en los paquetes. Dado que UDP no debe mantener ningún tipo de estado en los sistemas terminales, no hay casi sobrecarga en los hosts por el uso de UDP. Además, el cabezal de los segmentos UDP es bastante pequeño en comparación con TCP, por lo que el overhead en el envío de los datos es poco.

\* Se puede enviar a la mayor tasa posible permitida por la red. Dado que UDP no implementa control de congestión, UDP no agrega ninguna limitante a la velocidad a la que se pueden transmitir los datos.

Desventajas:

\* No hay ninguna garantía de que los datos lleguen al receptor. Dado que UDP no implementa transferencia fiable, no es posible asegurar que los datos enviados hayan sido recibidos por el receptor.

\* Es posible saturar al receptor o incluso saturar la red y no darse cuenta del problema. Dado que UDP no implementa control de flujo ni control de congestión, es posible que se generen problemas como ser la pérdida de paquetes debido a que el receptor no puede procesar los datos tan rápido como son recibidos o que incluso se sature la capacidad de la red. En este caso, la aplicación experimentaría una pérdida considerable de paquetes.

b) DNS es un ejemplo de una aplicación que utiliza de forma preferente UDP (aunque puede utilizar TCP en algunas ocasiones). Según la propia RFC de DNS: los datagramas son preferidos para las 'queries' debido a su bajo overhead y mejor performance. A esto se podría agregar que los mensajes DNS son en general pequeños y además la pérdida de un mensaje no es algo grave ya que se puede reenviar con facilidad.

#### **Pregunta 4 (8 puntos)**

Considere el protocolo de transferencia de datos fiable RDT 2.2:

- a) ¿Qué problemas pueden surgir cuando se utiliza este protocolo en un canal que puede corromper y perder paquetes?
- b) ¿Qué se debe agregar/modificar a este protocolo para que pueda realizar la transferencia fiable en un canal que puede corromper y perder paquetes? Justifique.

#### **Solución**

- a) El protocolo RDT 2.2 está diseñado para funcionar en un canal que puede corromper los paquetes pero no los puede perder.  
Por esto, asume que los mensajes siempre llegarán. Esto genera que el emisor en cualquier estado siempre espera por datos de la capa superior o un mensaje de reconocimiento (positivo o negativo) de la capa inferior.  
En el caso que un mensaje se pierda, esto generará que el emisor pueda quedar siempre en el estado de esperando un mensaje de la capa inferior y nunca salga de ese estado.
- b) El protocolo RDT 2.2 está diseñado para funcionar en un canal que puede corromper los paquetes pero no los puede perder.  
Por esto, asume que los mensajes siempre llegarán. Esto genera que el emisor en cualquier estado siempre espera por datos de la capa superior o un mensaje de reconocimiento (positivo o negativo) de la capa inferior.  
Para que pueda ser utilizado en un canal con pérdidas se debe agregar temporizadores en el emisor. De esta forma, ante la pérdida de un mensaje (o su reconocimiento) el emisor luego de un tiempo razonable inferirá que el mensaje se perdió y realizará el reenvío correspondiente.

#### **Pregunta 5 (12 puntos)**

La computadora A necesita enviar un mensaje que consta de 9 paquetes a la computadora B, utilizando un protocolo de ventana deslizante de tamaño 3 y Go-Back-N como estrategia de transporte

