

Redes de Computadoras
Solución - Examen – 18 de diciembre 2017
(ref: solredes20171218.odt)

Instrucciones

- Indique su nombre completo y número de cédula en cada hoja.
- Numere todas las hojas e indique la cantidad total de hojas que entrega en la primera.
- Escriba las hojas de un solo lado y utilice una caligrafía claramente legible.
- Comience cada pregunta teórica y cada ejercicio en una hoja nueva.
- Sólo se responderán dudas de letra. No se responderán dudas de ningún tipo los últimos 30 minutos del examen.
- El examen es individual y sin material. Apague su teléfono celular mientras esté en el salón del examen.
- Es obligatorio responder correctamente al menos 15 puntos en las preguntas teóricas y 20 de los problemas prácticos. Los puntos ganados en el curso se suman a los puntos de teórico.
- El puntaje mínimo de aprobación es de 60 puntos.
- Para todos los ejercicios, si es necesario, puede suponer que dispone de los tipos de datos básicos (p.ej. lista, cola, archivo, string, etc.) y sus funciones asociadas (ej: tail(lista), crear(archivo), concatenar(string, string)).
- Justifique todas sus respuestas.
- Duración: 3 horas. Culminadas las 3 horas el alumno no podrá modificar las hojas a entregar de ninguna forma.

Preguntas Teóricas

Pregunta 1 (10 puntos)

Considerando el conjunto de protocolos y servicios globalmente conocidos como “correo electrónico” se pide:

- a) Identifique los *endpoints* involucrados y protocolos utilizados en el envío de un correo electrónico.
- b) Identifique los *endpoints* involucrados y protocolos utilizados en la descarga de un correo electrónico.
- c) Explique el rol que cumple el protocolo HTTP en el contexto del correo electrónico actual, describiendo el concepto de “webmail”. Describa específicamente como complementa o reemplaza los protocolos mencionados en a) y b).

Solución Pregunta 1

a) Para el envío de correo electrónico se utiliza el protocolo SMTP. El cliente de correo (p.e. Thunderbird, K9, Outlook, mail, elm, etc.) implementa el endpoint utilizado por el usuario, mientras que, el servidor de correo del dominio del cliente implementa el endpoint al que se conecta el cliente. Posteriormente, este servidor se conecta con el servidor del dominio del receptor del correo, y en esta comunicación los endpoints están implementados por los servidores (Postfix, qmail, sendmail, etc.).

b) Para la descarga de correo se pueden utilizar tanto IMAP como POP. En ambos casos, una aplicación cliente (Thunderbird, Eudora, Pegasus mail, pine, etc.) será la responsable implementar su endpoint, mientras que del lado servidor una implementación de dicho protocolo será la responsable de implementar dicho endpoint (cyrus, dovecot, etc.)

c) Por muchos motivos se eligió evitar que el usuario final requiriera la instalación de software en su PC para la descarga de correo y se decidió permitir que éste pueda “ver” el correo sin necesidad de descargarlo. Allí surge el webmail. Para el envío de correo, desde un servidor web se ofrece una interfaz de edición (HTML). Luego de finalizar la escritura del correo, la aplicación asociada al servidor web es quién se conecta con el servidor y habla SMTP con éste. De allí en más el proceso es el descrito en a). Entre el servidor web y navegador del cliente se implementa el protocolo HTTP, siendo un navegador (firefox, chromium, opera, etc.) y un servidor web (apache, nginx, etc.) los endpoints. Para la comunicación entre la aplicación que corre en el servidor y el servidor de correo probablemente se utilice alguna biblioteca del lenguaje de programación de dicha aplicación para implementar el endpoint SMTP (p.e. php-mail o similares). En el caso relativo a la lectura de mail, la aplicación servidora implementa POP o IMAP contra un servidor de correo e implementa la visualización sobre HTML, que es vista por el cliente desde el navegador. El HTTP se utiliza para transportar la información (HTML) entre un navegador y una aplicación que implementa los protocolos de correo específicos.

Pregunta 2 (10 puntos)

- a) Explique el mecanismo de control de congestión de TCP (Transmission Control Protocol).
- b) Respecto al control de congestión, ¿en qué difiere la versión TCP Reno de la versión TCP Tahoe?

Solución Pregunta 2

a) TCP emplea un mecanismo de control de congestión terminal a terminal, donde cada emisor limita la tasa de transmisión de datos a través de su conexión en función de la congestión de red percibida. Para ello, cada emisor, mantiene una variable “ventana de congestión” que impone una restricción a la velocidad a la que TCP puede enviar tráfico a la red: la cantidad de datos no reconocidos en un emisor no puede exceder el mínimo entre la “ventana de congestión” y la “ventana de recepción”.

Redes de Computadoras

La percepción de congestión en la red por parte de un emisor TCP se basa en la ocurrencia de "sucesos de pérdida": fin de temporización o 3 ACKs duplicados relativos a la conexión en cuestión.

El ajuste del tamaño de la "ventana de congestión" se hace en función de los ACKs que se reciben.

En TCP:

Un segmento perdido implica congestión y, por tanto, la velocidad del emisor TCP debe reducirse cuando ello ocurre.

Un segmento que ha sido reconocido indica que la red está entregando los segmentos del emisor al receptor y, por tanto, la velocidad de transmisión del emisor puede incrementarse cuando llega un paquete ACK correspondiente a un segmento que todavía no había sido reconocido.

La estrategia de TCP para ajustar su velocidad de transmisión consiste en incrementar su velocidad en respuesta a la llegada de segmentos ACK hasta que ocurre un "suceso de pérdida", momento en el que reduce la velocidad de transmisión.

En funcionamiento por defecto de TCP, no existen notificaciones explícitas de congestión en la red, la misma se infiere.

El algoritmo de control de congestión de TCP consta de tres componentes fundamentales, llamadas: arranque lento, evitado de congestión y recuperación rápida.

En arranque lento, la "ventana de congestión" se establece en 1 MSS y se incrementa en 1 MSS cada vez que se reconoce un segmento transmitido ("crecimiento exponencial").

Cuando recibe 3 ACKs duplicados

Cuando termina un temporizador (inferencia de congestión en la red), la "ventana de congestión" pasa a valer 1 MSS, vuelve al arranque lento y "umbral" es igual a "ventana de congestión" / 2.

Cuando el valor de "ventana de congestión" alcanza el valor de "umbral", TCP pasa de fase arranque lento a fase de evitado de congestión.

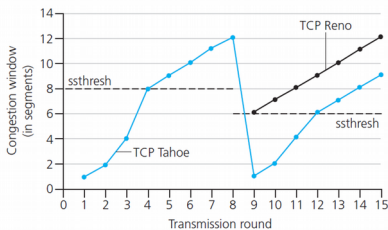
En la fase de evitado de congestión, TCP adopta un mecanismo más conservador de incrementado del valor de "ventana de congestión", pasando de duplicarla en cada RTT a incrementarla en 1 MSS por cada RTT (crecimiento lineal).

Si TCP no implementa recuperación rápida, ante un suceso de pérdida durante la fase de evitado de congestión, la "ventana de congestión" pasa a valer 1 MSS y "umbral" es igual a "ventana de congestión" / 2.

Si TCP sí implementa recuperación rápida, durante la fase de evitado de congestión y si el suceso de pérdida es "3 ACKs duplicados", la "ventana de congestión" pasa a valer ("ventana de congestión" / 2)+3 (para contemplar los 3 ACKs que abandonaron la red).

Durante la fase de recuperación rápida el crecimiento de la ventana de congestión es lineal (1 MSS por cada ACK duplicado hasta que se recibe un ACK nuevo, momento en que retorna a la fase de evitado de congestión.

Durante la fase de recuperación rápida si ocurre un fin de temporización, pasa a la fase de arranque lento, con "ventana de congestión" = 1 MSS y "umbral" = "ventana de congestión" / 2.



b) TCP Tahoe siempre pasa a estado "arranque lento" ante cualquier "suceso de pérdida". TCP Reno incorpora el estado de "recuperación rápida".

Pregunta 3 (10 puntos)

- Explique al menos 4 ventajas de IPv6 sobre IPv4.
- Explique por qué aún no se ha producido el reemplazo total de IPv4 por IPv6.
- Explique por qué la implementación de IPv6 es más limpia que la de IPv4 desde el punto de vista de la aislación de capas de red y enlace.

Solución Pregunta 3

- Mayor espacio de direcciones
 - Incorpora servicios de autoconfiguración (SAAC)
 - Procesamiento de cabecales (y extensiones) más eficiente
 - Elimina características no utilizadas en IPv4 (ToS)
 - Posibilidad de contar con cargas útiles superiores a los 64K (Jumbograms)
 - Soporte para etiquetado de flujo (Flow label)
- IPv6, como protocolo, se considera maduro desde el año 1998, y ha estado funcionando sobre Internet desde entonces, sin haber llegado hasta el usuario final. Para que el protocolo pueda utilizarse entre cualquier par de hosts tiene que implementarse en toda Internet, siendo este el motivo principal que ha frenado el despliegue global: tienen que cambiarse las implementaciones de todos los routers, de todos los equipos de acceso y de todos los sistemas operativos de los clientes. También muchos protocolos (que no implementan adecuadamente la separación entre capas) deben ser revisados para que funcionen con el nuevo protocolo de capa

Redes de Computadoras

de red. El cambio se ha venido posponiendo mientras se ha podido preservar la inversión en infraestructura.

c) La adaptación de IPv4 al enlace no fue considerada en su totalidad al momento de su estandarización. Por ejemplo para la adaptación a la familia de protocolos Ethernet fueron requeridos nuevos protocolos como ARP y RARP, entre otros. En IPv6 el protocolo de control (ICMPv6) fue definido con estas necesidades en mente, por lo que, la adaptación es realizada por el mismo protocolo, de una forma clara y definida. Siempre se usan mensajes IPv6 para que IPv6 funcione, sin requerir de convenciones poco claras o de protocolos auxiliares por fuera de la suite.

Pregunta 4 (10 puntos)

- a) Describa las características fundamentales del protocolo FTP (File Transfer Protocol).
- b) Mencione y explique el uso los principales comandos de FTP.

Solución Pregunta 4

a) FTP es un protocolo que permite la transferencia de uno o más archivos entre dos nodos. Está implementado utilizando el modelo cliente/servidor, siendo el cliente quien inicia la solicitud de transmisión, desde o hacia el servidor.

Este protocolo utiliza en su implementación dos conexiones TCP: control (puerto 21) y datos (puerto 20). Por ello se dice que el control de la conexión es "fuera de banda".

Secuencia típica básica:

- El cliente establece una conexión al puerto 21 TCP del servidor.
- El acceso del cliente al servidor es controlado a través de esta conexión mediante el suministro de credenciales: usuario y contraseña.
- El cliente puede "recorrer" la estructura de directorios -y conocer su contenido- mediante comandos enviados por esta conexión, y recibiendo las respuestas correspondientes a través de la misma.
- Cuando el servidor recibe la petición para la transferencia de un archivo (o varios) en un sentido u otro, él establece una segunda conexión TCP con el cliente, desde su puerto 20. A través de ella se envía el archivo (o los archivos) y luego la misma se cierra.

b) Los mencionados en el curso:

USER nombre_de_usuario: nombre de usuario para autenticación

PASS: password del usuario para autenticación

LIST: contenido de un directorio

RETR nombre_de_un_archivo: transferencia de un archivo, desde el directorio actual en el servidor, hacia el cliente

STOR nombre_de_un_archivo: transferencia de un archivo hacia el directorio actual del servidor

Otros comandos posibles: pwd, cd, delete, mkdir, rmdir, mget, mput, chmod, lcd

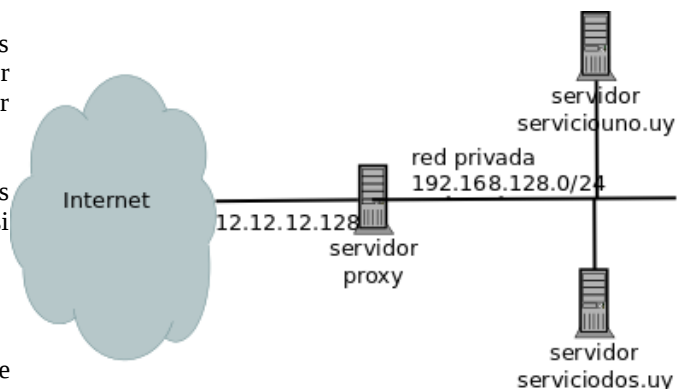
Redes de Computadoras
Problemas Prácticos

Problema 1 (30 puntos)

Sea la red mostrada en la figura, donde un equipo *servidor proxy* recibe conexiones web desde internet (puerto 80) y conecta con equipos internos a la red, configurados con direcciones IP privadas. En internet, el DNS devuelve la IP pública del *proxy* (12.12.12.128) para estos dos nombres de dominio www.serviciouno.uy y www.serviciodos.uy.

El *servidor proxy* debe recibir y procesar los accesos a los dos dominios mencionados. Como los dominios son atendidos por diferentes servidores, se deben redirigir los pedidos al servidor que corresponda.

El *servidor proxy* debe funcionar de manera transparente, es decir, los clientes deberán poder usar el servicio web como si estuviesen directamente conectados a los servidores finales.



Se pide:

- Indique de que forma el *servidor proxy* puede determinar a que servidor desea acceder el cliente.
- Implemente en un lenguaje de alto nivel, utilizando las primitivas de *sockets*, el programa que ejecuta el *servidor proxy*. El *servidor proxy* deberá ser capaz de atender múltiples clientes simultáneamente. Defina claramente (pero no implemente) las funciones que utilice para parsear los mensajes HTTP.

Solución Problema 1

a) El servidor proxy utilizará el campo HOST del cabezal del mensaje HTTP.

```
b)
void proxy() {

    int serverSock = new socket();
    bind (serverSock, localhost, 80);

    while (true){
        int clientSock = accept(serverSock); //espero conexiones
        if (clientSock < 0)
            break;
        thread.new (atenderCliente, clientSock); //nuevo hilo para el cliente
    }
    close (serverSock);
}
}
```

```
void atenderCliente (clientSock) {

    int error;
    String targetHost=NULL;
    String object = "";

    //leemos hasta encontrar campo host
    while (targetHost==NULL) {
        String fragment;
        error = recv (clientSock, fragment);
        if (error < 0){
            close (clientSock)
            exit (-1);
        }
        object+=fragment;
        targetHost = getHostFromHTTP (object);
    }

    int serverSock = new socket();

    if ( targetHost == "www.serviciouno.uy"){
```

Redes de Computadoras

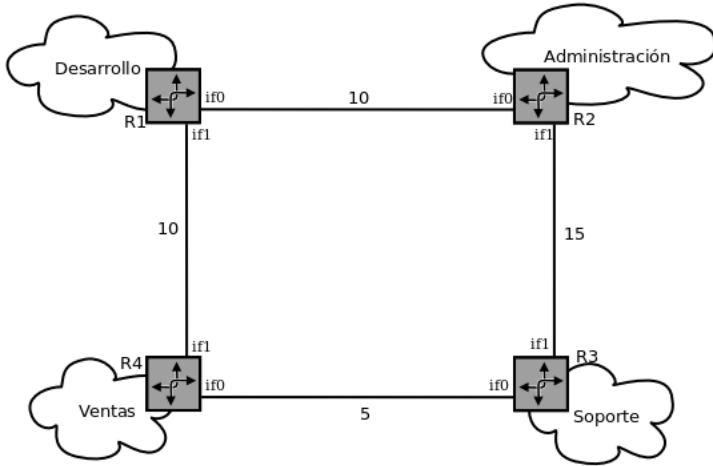
```
error = connect (serverSock, getIP ("serviciouno.uy"), 80);
if (error < 0){
    close (clientSock)
    exit (-1);
}
}
elseif (targetHost=="www.serviciodos.uy"){
    error = connect (serverSock, getIP ("serviciodos.uy"), 80);
    if (error < 0){
        close (clientSock)
        exit (-1);
    }
} else {
    close (clientSock)
    exit (-1);
}

//envío los datos recibidos hasta el momento
error = send (serverSock, object);
if (error < 0){
    close (serverSock);
    close (clientSock);
    exit (-1);
}

//hilos para enviar datos y esperar respuesta de los servidores y del cliente
thread.new (pushData, clientSock, serverSock);
thread.new (pushData, serverSock, clientSock);
}

void pushData(Socket src, Socket dst)
String data;
while (true){
    error = recv (src, data);
    if (error < 0)
        break;
    error = send (dst, data);
    if (error < 0)
        break;
}
close (src);
close (dst);
}
```

Problema 2 (30 puntos)



La figura representa la red de una empresa que tiene cuatro secciones en pisos diferentes de un edificio.

Se dispone del prefijo de red 100.100.100.0/22, y se desea planificar la conectividad instalando cuatro routers (R1, R2, R3 y R4) y asignando direcciones a cada sección, teniendo en cuenta que se necesitan los siguientes puestos de trabajo:

- Administración nunca superará los 40.
- Soporte tiene 3 puestos fijos pero necesita otros 5 para pruebas.
- Ventas necesitará hasta 150.
- Desarrollo tiene 200 puestos, y además corre los servidores de la empresa (archivos, DNS, DHCP, etc) que no superan los 10.

Se pide:

- Asigne prefijos a las subred de cada sección y a los enlaces.
- Escriba las tablas de *forwarding* de los routers para tener conectividad total entre las secciones de la empresa. A los efectos de determinar dichas tablas, se sabe que los routers tienen la capacidad de balancear la carga entre caminos que tengan igual costo. Utilice el formato Prefijo de Destino | Interfaz de salida | Next-Hop.
- Considere ahora el siguiente caso: los gerentes estudiaron la solución propuesta por los técnicos, y decidieron que no se justifica la inversión en routers, y por lo tanto se deben reemplazar por switches. Determine los cambios necesarios en la configuración para soportar este cambio.
- Los switches tienen la capacidad de determinar un árbol de cubrimiento mínimo a los efectos de evitar ciclos, y soportan autoaprendizaje. Suponiendo que dicho árbol está establecido, y que las tablas de los switches están vacías, determine la secuencia de acciones que desencadena en cada switch el envío de una trama desde un puesto de trabajo en Soporte (MAC_S1) a un puesto de trabajo en Administración (MAC_A1).

Solución Problema 2

a)
 Las necesidades de direcciones son las siguientes:
 Administración -> 40 puestos + 1 interfaz de router + red + broadcast < 64 => /26
 Soporte -> 8 puestos + 1 interfaz de router + red + broadcast < 16 => /28
 Ventas-> 150 puestos + 1 interfaz de router + red + broadcast < 256 => /24
 Desarrollo-> 210 puestos + 1 interfaz de router + red + broadcast < 256 => /24
 Enlaces punto a punto -> 2 interfaces + red + broadcast =4 => /30

Se puede resolver asignando:
 Desarrollo -> 100.100.100.0/24
 Dejamos libre el prefijo 100.100.101.0/24 para posibles crecimientos de Desarrollo (no lo pide la letra, es una buena práctica)
 Ventas-> 100.100.102.0/24
 Administración -> 100.100.103.0/26
 Soporte -> 100.100.103.128/28
 Enlaces punto a punto-> 100.100.103.240/30 (R1-R2), 100.100.103.244/30 (R2-R3), 100.100.103.248/30 (R3-R4), 100.100.103.252/30 (R4-R1)
 Tomamos la dirección más baja de cada subred para la interfaz correspondiente del router mencionado en primer término.

b)
 Nota: se nombran como "if2" a las interfaces de los routers conectados a las redes de cada sección.

Redes de Computadoras

R1

Prefijo de Destino	Int. de salida	Next-Hop	Costo (Opcional)
100.100.100.0/24 (Desarrollo)	if2	Directamente Conectado	0
100.100.103.0/26 (Administración)	if0	100.100.100.242 (R2-if0)	10
100.100.102.0/24 (Ventas)	if1	100.100.100.253 (R4-if1)	10
100.100.103.128/28 (Soporte)	if1	100.100.100.253 (R4-if1)	15
100.100.103.240/30 (R1-R2)	if0	Directamente Conectado	0
100.100.103.244/30 (R2-R3)	if0	100.100.100.242 (R2-if0)	10
100.100.103.248/30 (R3-R4)	if1	100.100.100.253 (R4-if1)	10
100.100.103.252/30 (R4-R1)	if1	Directamente Conectado	0

R2

Prefijo de Destino	Int. de salida	Next-Hop	Costo (Opcional)
100.100.100.0/24 (Desarrollo)	if1	100.100.100.241 (R1-if0)	10
100.100.103.0/26 (Administración)	if2	Directamente Conectado	0
100.100.102.0/24 (Ventas)	if0	100.100.100.241 (R1-if0)	20
100.100.102.0/24 (Ventas)	if1	100.100.100.246 (R3-if1)	20
100.100.103.128/28 (Soporte)	if1	100.100.100.246 (R3-if1)	15
100.100.103.240/30 (R1-R2)	if0	Directamente Conectado	0
100.100.103.244/30 (R2-R3)	if1	Directamente Conectado	0
100.100.103.248/30 (R3-R4)	if0	100.100.100.241 (R1-if0)	10
100.100.103.252/30 (R4-R1)	if1	100.100.100.246 (R3-if1)	15

R3

Prefijo de Destino	Int. de salida	Next-Hop	Costo (Opcional)
100.100.100.0/24 (Desarrollo)	if0	100.100.100.250 (R4-if0)	15
100.100.103.0/26 (Administración)	if1	100.100.100.245 (R2-if1)	15
100.100.102.0/24 (Ventas)	if0	100.100.100.250 (R4-if0)	5
100.100.103.128/28 (Soporte)	if2	Directamente Conectado	0
100.100.103.240/30 (R1-R2)	if0	100.100.100.250 (R4-if0)	15
100.100.103.240/30 (R1-R2)	if1	100.100.100.245 (R2-if1)	15
100.100.103.244/30 (R2-R3)	if1	Directamente Conectado	0
100.100.103.248/30 (R3-R4)	if0	Directamente Conectado	0
100.100.103.252/30 (R4-R1)	if0	100.100.100.250 (R4-if0)	5

R4

Prefijo de Destino	Int. de salida	Next-Hop	Costo (Opcional)
100.100.100.0/24 (Desarrollo)	if1	100.100.100.254 (R1-if1)	10
100.100.103.0/26 (Administración)	if0	100.100.100.249 (R3-if0)	20
100.100.103.0/26 (Administración)	if1	100.100.100.254 (R1-if1)	20
100.100.102.0/24 (Ventas)	if2	Directamente Conectado	0
100.100.103.128/28 (Soporte)	if0	100.100.100.249 (R3-if0)	5
100.100.103.240/30 (R1-R2)	if1	100.100.100.254 (R1-if1)	10
100.100.103.244/30 (R2-R3)	if0	100.100.100.249 (R3-if0)	5
100.100.103.248/30 (R3-R4)	if0	Directamente Conectado	0
100.100.103.252/30 (R4-R1)	if1	Directamente Conectado	0

Redes de Computadoras

c)
Ahora pasamos a tener una red plana (una sola subred o LAN), y por lo tanto lo único que hay que cambiar con las máscaras de subred, todos los *hosts* pasarán a tener máscara /22. Se eliminan las direcciones IP de las interfaces de los enlaces punto a punto que pasan a ser puertos puros de capa 2.

d)
El spanning tree "mata" el enlace de costo 15.
Por lo tanto una trama desde Soporte a Administración será difundida (*flooding* controlado) por cada uno de los switches (S3, S4, S1, S2) en secuencia, y en cada reenvío insertarán una entrada para la MAC_S1 en el puerto de entrada correspondiente, usando el mecanismo de *self-learning* de los *switches*.