

Redes de Datos

Licenciatura en Sistemas

Laboratorio 2

“Capa de Aplicación”

1. Objetivos

En este laboratorio se verán algunos protocolos de capa aplicación, su utilización y sus principales características.

Los protocolos a estudiar serán los siguientes:

- Telnet
- FTP
- SMTP
- POP3
- HTTP
- SSH

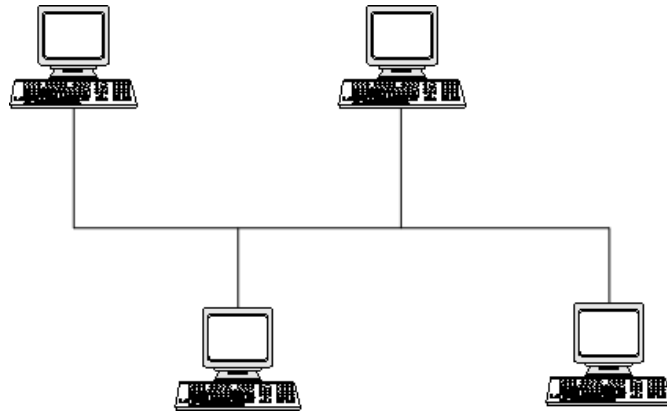
2. Preparación

Es importante entender el direccionamiento en las diferentes capas. Para que dos aplicaciones sean capaces de comunicarse necesitan:

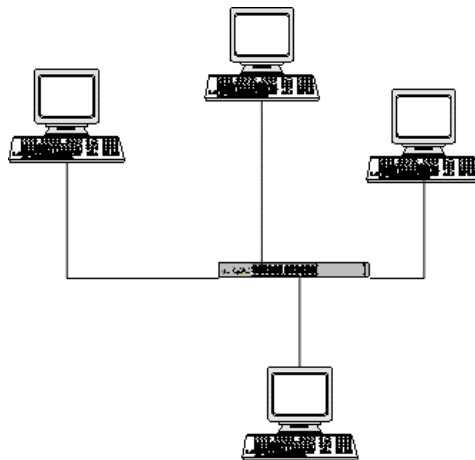
- Conocer las direcciones de capa de red: nombre o dirección IP
Ej: 200.40.30.163 o correoadinet.adinet.com.uy
- Conocer la dirección de capa de transporte o puerto: accedemos a los servicios en capa de aplicación conectándonos por un numero de puerto en la capa de Transporte por el cual se atienden los pedidos de conexión para dicha aplicación
Ej: servidor web puerto 80 TCP

Sin entrar en detalle haremos una breve descripción sobre los elementos que nos interconectan en una red de área local (LAN).

Normalmente en una LAN las computadoras (o hosts) se interconectan utilizando una tecnología de difusión llamada Ethernet (IEEE 802.3). Originalmente la conexión física de la red era en “bus”, todos los equipos compartían un mismo cable (cable coaxial) por el cual lo que transmitía cualquier equipo era escuchado por todos los demás equipos conectados.



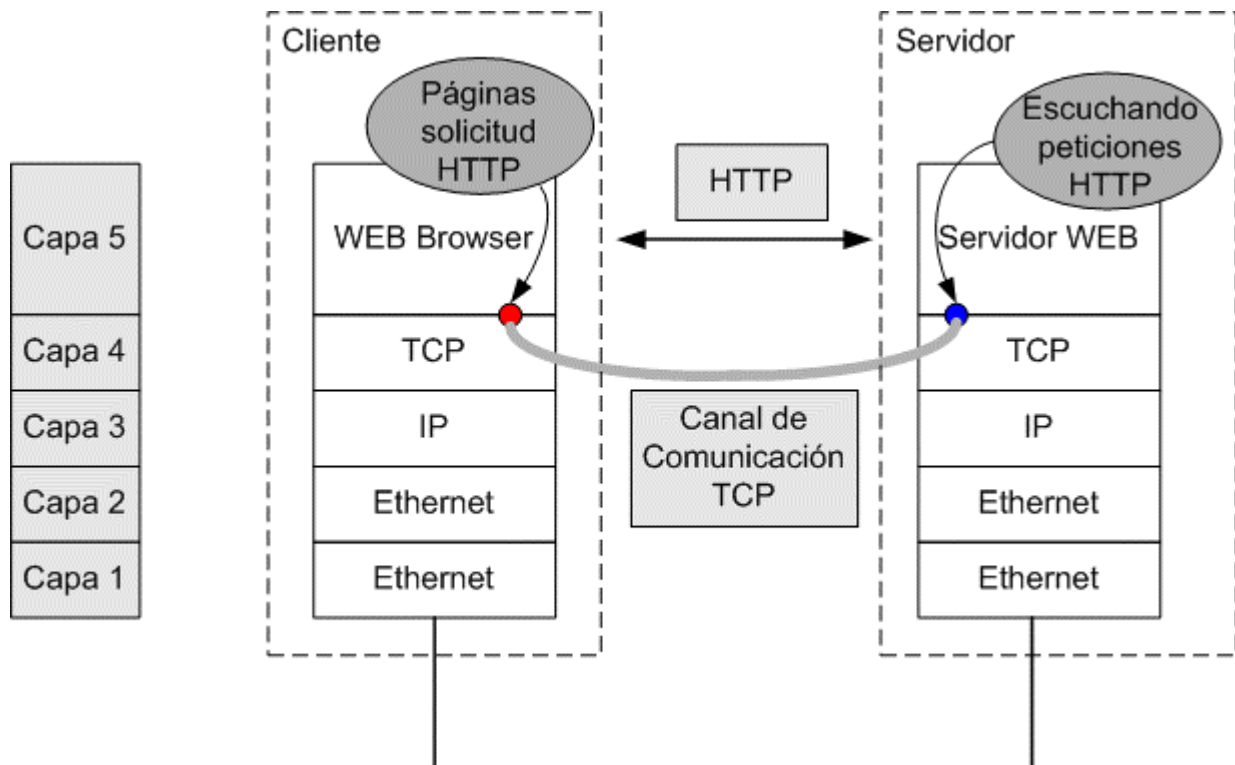
Posteriormente se comenzó a utilizar el cable UTP (Unshielded Twisted Pair o Par Trenzado sin Blindaje) junto con equipos concentradores llamados “hub”. Esto modificó el modelo de conexión física de todos conectados al mismo cable a una conexión estrella donde en el centro está el concentrador. Si bien se modificó la conexión física, el funcionamiento lógico de la conexión continúa siendo en “bus” ya que un hub repite toda la información ingresada por un puerto en los puertos restantes.



El cable UTP, está conformado por 8 cables de cobre llamados hilos trenzados de a dos, a estos se les agrega una protección exterior que los mantiene unidos. En las puntas se utilizan conectores RJ 45 que son parecidos a los de teléfono (RJ 11) pero de tamaño mayor.

En una LAN podemos encontrar otros tipos de equipos que nos permiten realizar la tarea de concentración de PC llamados “switches”. Los mismos ofrecen mejoras a nivel de capacidad de comunicación entre los equipos (más bits por segundo), permitiendo la mayoría del tiempo que dos equipos perciban como si estuviesen solos (disponibilidad de toda la capacidad para ellos). No obstante se continúan permitiendo algunos tipos de difusión.

A continuación continuaremos con una breve descripción de cómo se interrelacionan los conceptos vistos en clase para una conectividad entre una aplicación cliente y una aplicación servidor.



El ejemplo ilustra el funcionamiento de un protocolo de uso cotidiano, el mismo consiste en un usuario que mediante la utilización de un browser (User-Agent) desea conectarse al servidor web y traer la página principal, normalmente digita <http://www.empresa.com.uy>. Para poder realizar esto primero se establece un canal de comunicación a nivel de capa de Transporte utilizando TCP. Para poder realizar este canal de comunicación necesitamos IP_c la dirección ip del cliente (conocida por el), IP_s dirección ip del servidor (normalmente obtenida mediante una consulta a un servicio de nombres), puerto en donde escucha el servidor web (normalmente puerto 80) y un puerto de origen en el cliente (usualmente elegido por el sistema operativo). Una vez establecido el canal de comunicación el cliente realiza una petición de la pagina web inicial (normalmente index.htm) utilizando mensajes definidos en el protocolo http, el servidor web reconoce estos mensajes y responderá con la solicitud realizada utilizando también mensajes del protocolo http. En este momento el cliente posee localmente una copia del index.htm, por lo que su browser se dispone a presentarlo en la pantalla.

3. Telnet

El protocolo permite abrir, en el equipo sobre el que se esta trabajando, una terminal remota. En otras palabras, nos permitirá establecer una conexión a un equipo remoto, obteniendo una terminal, mediante la cual podremos ejecutar comandos sobre el equipo al que accedemos.

Para poder hacer uso de esta aplicación, es necesario que el equipo remoto al que se quiere acceder soporte dicho protocolo (debe tener activo un *servidor Telnet*) y el equipo desde el cual deseamos acceder deberá disponer de un *cliente Telnet*.

Observamos que se puede dar el caso particular de que el equipo origen y destino coincidan; es decir, es posible “hacer Telnet” al propio equipo, siempre y cuando este cumpla con las características de cliente y de servidor mencionadas.

El formato básico del comando para invocar el protocolo TELNET es el siguiente:

```
telnet <equipo> <puerto>
```

Opción	Función
<equipo>	Equipo al que se desea acceder
<puerto>	Puerto de dicho equipo al que se desea acceder

Para indicar el equipo al que se desea acceder es necesario especificar la dirección de capa de Transporte del equipo destino, indicando la dirección de capa 3 y el número de puerto. Por ejemplo, si queremos acceder al puerto 25 (capa 4) de un equipo cuya dirección IP (capa 3) es la 192.168.1.1, entonces el comando a utilizar tendrá la siguiente sintaxis:

```
telnet 192.168.1.1 25
```

De esta forma, se abrirá directamente una conexión al puerto 25 del equipo remoto cuya dirección IP es la 192.168.1.1.

Se observa que es posible especificar un nombre de dominio en lugar de una dirección IP para identificar al equipo, pero esto requiere que el equipo origen tenga capacidad de resolver dicho nombre de dominio (el protocolo de resolución de nombre de dominio “DNS” se explicará más adelante en el curso).

El comando “Telnet” admite también una serie de modificadores y de opciones que en general podremos ver utilizando la ayuda del sistema operativo:

Sistema Operativo	Comando
Windows	telnet -h
Linux	man telnet

4. SMTP

El protocolo para envío de correos electrónico SMTP (Simple Mail Transport Protocol) se encarga de transferir mensajes entre el usuario generador (From: o De:) del mensaje a otro usuario destinatario del mensaje (To: o Para:).

Este protocolo utiliza el conocimiento adquirido del sistema postal, encontramos entonces que el protocolo SMTP en su estándar inicial RFC 821, define varias partes en un correo, una parte inicial equivalente al sobre postal, luego una parte que contiene líneas de Encabezado que terminan en una línea en blanco y por ultimo el cuerpo del mensaje. Las líneas de encabezado y el cuerpo del mensaje se escriben a continuación del sobre en ese orden.

Ejemplo:

Concepto	Comandos	Funciones
Sobre	<pre> helo hermes.ort.edu.uy mail from:<docente@ort.edu.uy> rcpt to:<alumno@adinet.com.uy> data </pre>	El sobre esta conformado por el comando helo, “mail from:” y “rcpt to:” Esta información no es visible por el usuario solo se utiliza para “encaminar” los correos.
Encabezado	<pre> From: “El docente” <docente@ort.edu.uy> To: “El alumno” <alumno@adinet.com.uy> Subject: Proxima Clase </pre>	<p>El encabezado se escribe inmediatamente luego del comando DATA.</p> <p>Fin del encabezado (línea en blanco)</p>
Cuerpo	<pre> Estimado alumno, La proxima clase realizaremos ejercicios practicos. Slds, Docente . </pre>	<p>Texto del mensaje</p> <p>Un punto como único carácter finaliza el cuerpo y con esto el mensaje</p>
Final	Quit	La sesión de envío de correo finaliza con esta línea.

Resumen de comandos de SMTP

Comandos	Función
helo host	Identifica al equipo que esta iniciando la conexión SMTP
ehlo host	Identifica al equipo que esta iniciando la conexión SMTP (ESMTP RFC 2821)
mail from: dirección	Identifica en el sobre al remitente
rcpt to: dirección	Identifica en el sobre al destinatario (puede haber varios)
Data	Comienza el cuerpo del mensaje, termina con un ”.” en la línea
Quit	Cierra la conexión
Rset	Reinicia la conexión (descarta todo lo realizado)
Help	Lista los comandos SMTP aceptados

Cabe aclarar que cuando nos referimos a comandos del SMTP nos estamos refiriendo a los mensajes que intercambian información las entidades de capa de aplicación. Es importante aclarar que el SMTP solo soporta caracteres ASCII en todos los campos. Existen especificaciones mas modernas del protocolo SMTP (RFC 2821) y del formato de los Mensajes (encabezado + cuerpo, RFC 2822).

Dentro de los datos (área entre el comando SMTP data y una línea con un “.”) se reconocen dos regiones: encabezado (“header”) y el cuerpo (“body”). El cuerpo es de formato libre (cumpliendo con ASCII), es aquí donde se aparece lo que un usuario normalmente escribe. Por el contrario el encabezado posee una estructura bien definida, el mismo es utilizado para el transporte del correo y por los agentes de correo.

La siguiente tabla es un extracto de los encabezados de SMTP mas comunes.

Encabezado:	Significado
To:	Dirección(es) de correo de destinatario(s) primario(s)
Cc:	Dirección(es) de correo de destinatario(s) secundarios(s)
Bcc:	Dirección(es) de correo de destinatario(s) ocultos(s)
From:	Nombre y dirección de la persona que escribió el mensaje
Sender:	Dirección de destino del remitente
Received:	Utilizado por los agentes de transporte a lo largo de la “ruta”
Return-Path	Pueden ser utilizado para indicar un camino diferente de retorno hacia el remitente

5. POP3

El protocolo POP3 (Post Office Protocol versión 3) se especifica en la RFC 1939, este protocolo permite a un usuario poder acceder a su servidor de correo y poder leer sus correos, “bajarlos”, borrarlos, etc.

La siguiente tabla muestra un resumen de comandos de POP3.

Comando	Función
USER dirección	Identifica al usuario ante el servidor POP
PASS contraseña	Envía la contraseña al servidor POP
STAT	Muestra la cantidad de mensajes y el espacio total en bytes.
LIST	Lista los mensajes identificados por número y tamaño en bytes.
RETR nro_mensaje	Muestra el contenido del mensaje cuyo número se indica.
DELE nro_mensaje	Elimina el mensaje cuyo número se indica.
QUIT	Cierra la conexión

6. HTTP

El protocolo HTTP (Hyper Transfer Transport Protocol) definido en RFC 1945, permite el intercambio de documentos de hipertexto, para ello HTTP define la estructura de los mensajes y el compartamiento del cliente (browser) y servidor (server) ante estos mensajes.

Una página web se compone de objetos, un objeto es un archivo, que puede ser del tipo HTML (Hyper Text Markup Language) , GIF, JPEG, Java Applet, MP3, etc; al cual se lo

puede acceder mediante una URL (Uniform Resource Locator). En general una URL se compone de dos partes: el host y el camino al archivo dentro del host, ejemplo:

www.empresa.cpm.uy/novedades/septiembre2005.html

Un navegador (o browser) realizara peticiones HTTP a los servidores por archivos HTML, una vez obtenidos verifican si dentro de ellos hacen referencias a otros objetos (como gifs), de ser asi realizan varias peticiones HTTP de los objetos faltantes (para poder desplegar en pantalla el contenido al usuario) a los servidores correspondientes.

La siguiente tabla muestra los comandos mas utilizados del protocolo HTTP.

Comando	Función
HEAD <URL> HTTP/1.0	Trae el encabezado de una página especificada en la URL. Requiere una línea en blanco y luego otro ENTER (dos ENTER en total)
GET <URL> HTTP/1.0	Trae el contenido del archive especificado en la URL. Requiere una línea en blanco y luego otro ENTER (dos ENTER en total)

HTTP 1.1 introduce mejoras al HTTP 1.0, se especifica en la RFC 2616.

7. FTP

El protocolo FTP (File Transfer Protocol) permite enviar o recibir archivos desde el servidor, se encuentra especificado en RFC 959. La gran diferencia que tiene con los protocolos anteriores es que maneja dos conexiones (dos puertos), una de ellas siempre permanece abierta y es por donde envío la información de control, por la otra conexión se realiza la transferencia de datos propiamente dicha. Dadas estas características no es posible emular la conexión utilizando el telnet.

En general todos los sistemas operativos implementan un cliente ftp el cual puede ser invocado escribiendo

```
ftp nombre_de_servidor_destino
```

se solicitara usuario y password para poder establecer la conexión de control, una vez establecida el comando help despliega la lista de comandos válidos.

8. SSH

El protocolo SSH (Secure SHell) es una programa de aplicación con un protocolo de aplicación asociado que agrega seguridad a la conexión. Su propósito es similar al del Telnet, poder ingresar a un equipo remoto utilizando usuario y password. La diferencia que posee con el Telnet es que provee una conexión segura con encriptación de la información (la encriptación se realiza en capa de aplicación). Este protocolo nos permitiría conectar dos máquinas que desean intercambiar información utilizando una infraestructura de red “poco confiable”.

No todos los sistemas operativos implementan un cliente ssh, para el caso de que se implemente o se logre instalar la aplicación para tal fin, el metodo general de invocación es

```
ssh nombre_de_servidor_remoto
```

normalmente de debe de especificar el usuario de alguna forma

```
ssh usuario@nombre_de_servidor_remoto  
ssh nombre_de_servidro_remoto -l usuario
```

luego se digita la contraseña.

Normalmente la primera vez que se relaiza un ssh entre dos equipos realizan un intercambio de clave sobre las cuales se pide confirmación, este detalle siempre es posible subsanarlo si se proporciona la clave por otro método (transportando un disquete) y se ingresa a los archivos de configuración adecuados.