Universidad de Guadalajara

Centro Universitario De Ciencias Exactas e Ingenierías

División de electrónica y computación

Departamento De Ciencias Computacionales

Ingeniería en Computación

Redes y Protocolos de Comunicación

Clave: I7031

Sección: D03

A.7: SUITE TCP / IP.

Alumno:

Arellano Granados Angel Mariano 218123444

Profesor: Anaya Oliveros Jorge

Fecha de Entrega: 23 – Marzo – 2022

Calificación:

Observaciones:

CONTENIDO:

INTRODUCCIÓN

Historia del Protocolo

La arquitectura de protocolos TCP/IP es resultado de la investigación y desarrollo llevados a cabo en la red experimental de conmutación de paquetes ARPANET, financiada por la Agencia de Proyectos de Investigación Avanzada para la Defensa (DARPA, Defense Advanced Research Projects Agency), y se denomina globalmente como la familia de protocolos TCP/IP. Esta familia consiste en una extensa colección de protocolos que se han especificado como estándares de Internet por parte de IAB (Internet Architecture Board).

OBJETIVOS:

GENERAL:

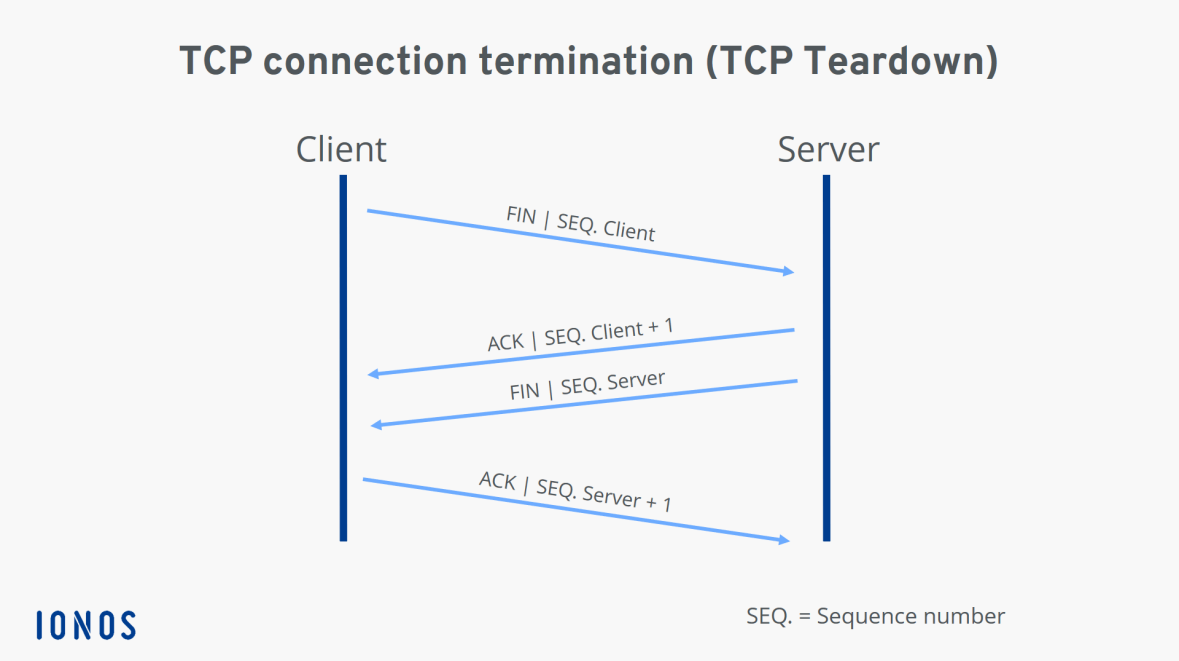
Conoces que es la Suite TCP e IP asi como su uso en las tecnologías de comunicación en la actualidad.

PARTICULAR:

Comprender a través de diagramas y listados las características, usos y metodologías del TCP / IP

DESARROLLO

Diagrama cliente / servidor



Formato de la Trama TCP

TCP utiliza un único tipo de unidad de datos de protocolo, llamado segmento TCP. La cabecera se muestra en la Figura 20.10. Ya que una cabecera debe servir para llevar a cabo todos los mecanismos del protocolo, ésta es más bien grande, con una longitud mínima de 20 octetos. Sus campos son los siguientes:

- **Puerto origen** (16 bits): usuario TCP origen.
- **Puerto destino** (16 bits): usuario TCP destino.
- **Número de secuencia** (32 bits): número de secuencia del primer octeto de datos en este segmento, excepto cuando está presente el indicador SYN. Si el indicador SYN está activo, se trata del número de secuencia inicial (ISN) y el primer octeto de datos es el ISN+1.
- **Número de confirmación** (32 bits): contiene el número de secuencia del siguiente octeto que la entidad TCP espera recibir.
- **Longitud de la cabecera** (4 bits): número de palabras de 32 bits de la cabecera.

- **Reservado** (6 bits): bits reservados para uso futuro. El RFC 3168 usa dos de esos bits para la función de notificación explícita de congestión. Una discusión sobre esta función está fuera de nuestro alcance.
- **Indicadores** (6 bits):
    - URG: el campo de puntero urgente es válido.
    - ACK: el campo de confirmación es válido.
    - PSH: función de forzado.
    - RST: reiniciar la conexión.
    - SYN: sincronizar los números de secuencia.
    - FIN: el emisor no enviará más datos.
- **Ventana** (16 bits): asignación de créditos para el control de flujo, en octetos. Contiene el número de octetos de datos, comenzando con el número de secuencia que se indica en el campo de confirmación que el emisor está dispuesto a aceptar.
- **Suma de comprobación** (16 bits): el complemento a uno de la suma modular complemento a uno de todas las palabras de 16 bits del segmento más una pseudocabecera.
- **Puntero urgente** (16 bits): este valor, cuando se suma al número de secuencia del segmento, contiene el número de secuencia del último octeto de la secuencia de datos urgentes. Esto permite al receptor conocer la cantidad de datos urgentes que llegan.
- **Opciones** (Variable): un ejemplo lo constituye la opción que especifica la longitud máxima de segmento que será aceptada.

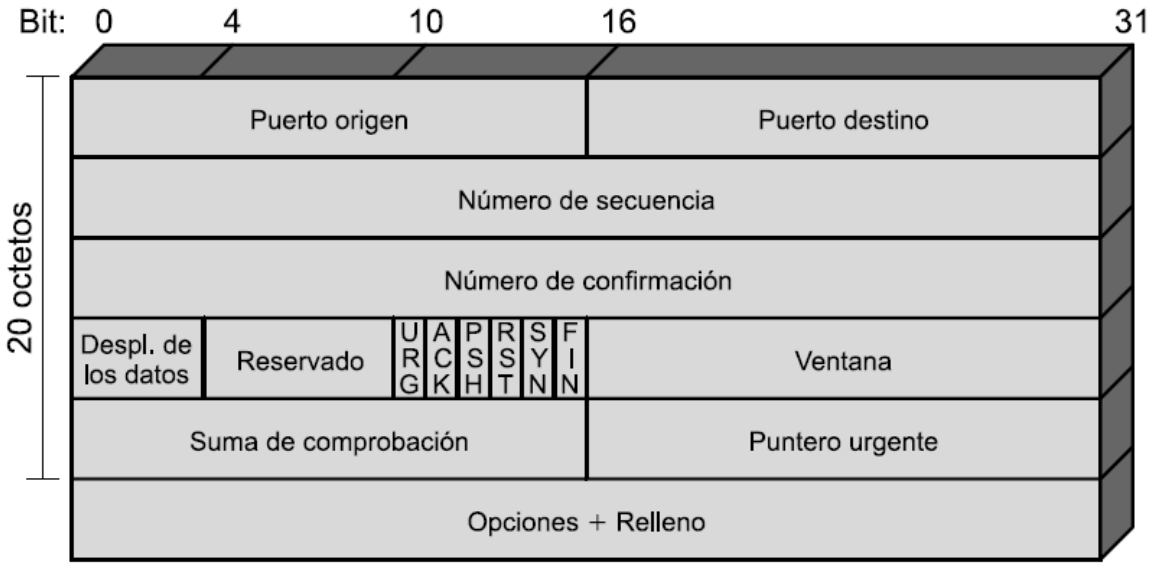


Figura 20.10. Cabecera de TCP.

Descripción

La Figura 2.13 muestra cómo se configuran los protocolos TCP/IP. Para poner de manifiesto que el conjunto total de recursos para la comunicación puede estar formado por varias redes, a dichas redes constituyentes se les denomina subredes.

Para conectar un computador a una subred se utiliza algún tipo de protocolo de acceso, por ejemplo, Ethernet. Este protocolo permite al computador enviar datos a través de la subred a otro computador o, en caso de que el destino final esté en otra subred, a un dispositivo de encaminamiento que los retransmitirá.

IP se implementa en todos los sistemas finales y dispositivos de encaminamiento. Actúa como un porteador que transportara bloques de datos desde un computador hasta otro, a través de uno o varios dispositivos de encaminamiento.

TCP se implementa solamente en los sistemas finales, donde supervisa los bloques de datos para asegurar que todos se entregan de forma fiable a la aplicación apropiada.

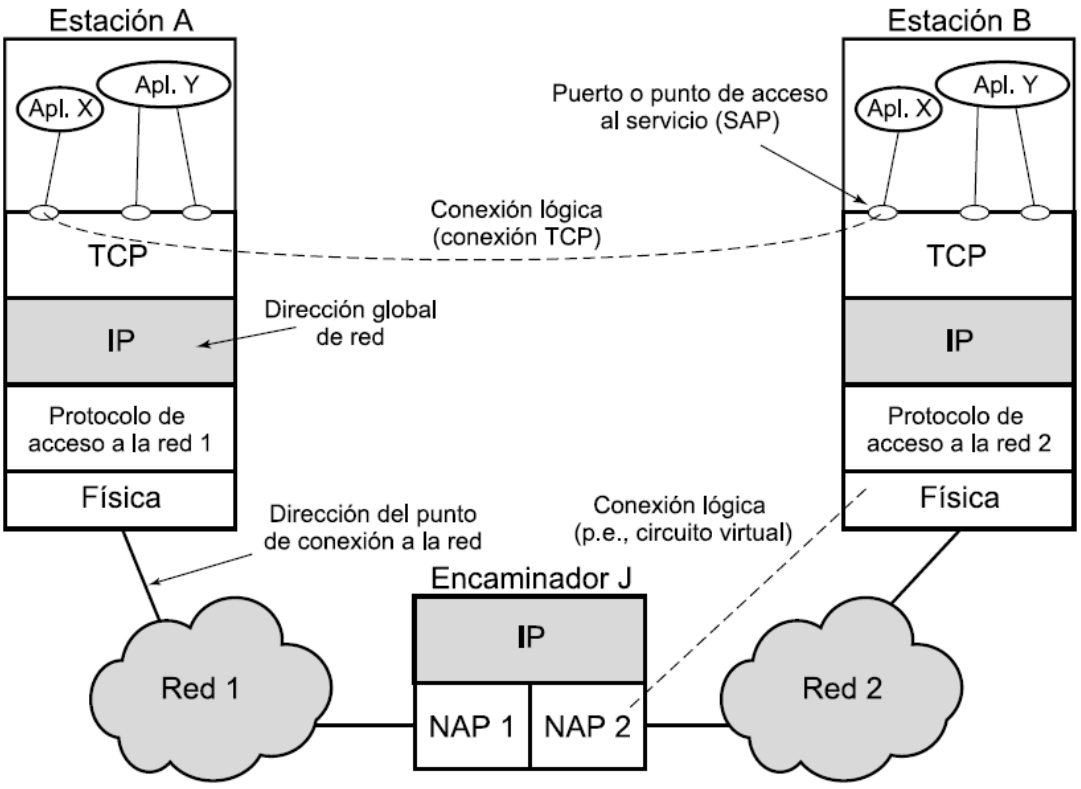


Figura 2.13. Conceptos de TCP/IP.

Supóngase que un proceso, asociado al puerto 1 en el computador A, desea enviar un mensaje a otro proceso, asociado al puerto 2 del computador B. El proceso en A pasa el mensaje a TCP con la instrucción de enviarlo al puerto 2 del computador B. TCP pasa el mensaje a IP con la instrucción de enviarlo al computador B. Obsérvese que no es necesario comunicarle a IP la identidad del puerto destino. Todo lo que necesita saber es que los datos van dirigidos al computador B. A continuación, IP pasa el mensaje a la capa de acceso a la red con el mandato expreso de enviarlo al dispositivo de encaminamiento J (el primer salto en el camino hacia B).

Para controlar esta operación se debe transmitir información de control junto con los datos de usuario. Supongamos que el proceso emisor genera un bloque de datos y lo pasa a TCP. TCP puede que divida este bloque en fragmentos más pequeños para hacerlos más manejables. A cada uno de estos fragmentos le añade información de control, denominada cabecera TCP, formando un segmento TCP. La información de control la utilizará la entidad par TCP en el computador B. Entre otros, en la cabecera se incluyen los siguientes campos:

- **Puerto destino**: cuando la entidad TCP en B recibe el segmento, debe conocer a quién se le deben entregar los datos.
- **Número de secuencia**: TCP numera secuencialmente los segmentos que envía a un puerto destino dado para que, si llegan desordenados, la entidad TCP en B pueda reordenarlos.
- **Suma de comprobación**: la entidad emisora TCP incluye un código calculado en función del resto del segmento. La entidad receptora TCP realiza el mismo cálculo y compara el resultado con el código recibido. Si se observa alguna discrepancia implicará que ha habido algún error en la transmisión.

A continuación, TCP pasa cada segmento a IP con instrucciones para que los transmita a B. Estos segmentos se transmitirán a través de una o varias subredes y serán retransmitidos en uno o más dispositivos de encaminamiento intermedios.

Esta operación también requiere el uso de información de control. Así, IP añade una cabecera de información de control a cada segmento para formar lo que se denomina un datagrama IP. En la cabecera IP, además de otros campos, se incluirá la dirección del computador destino.

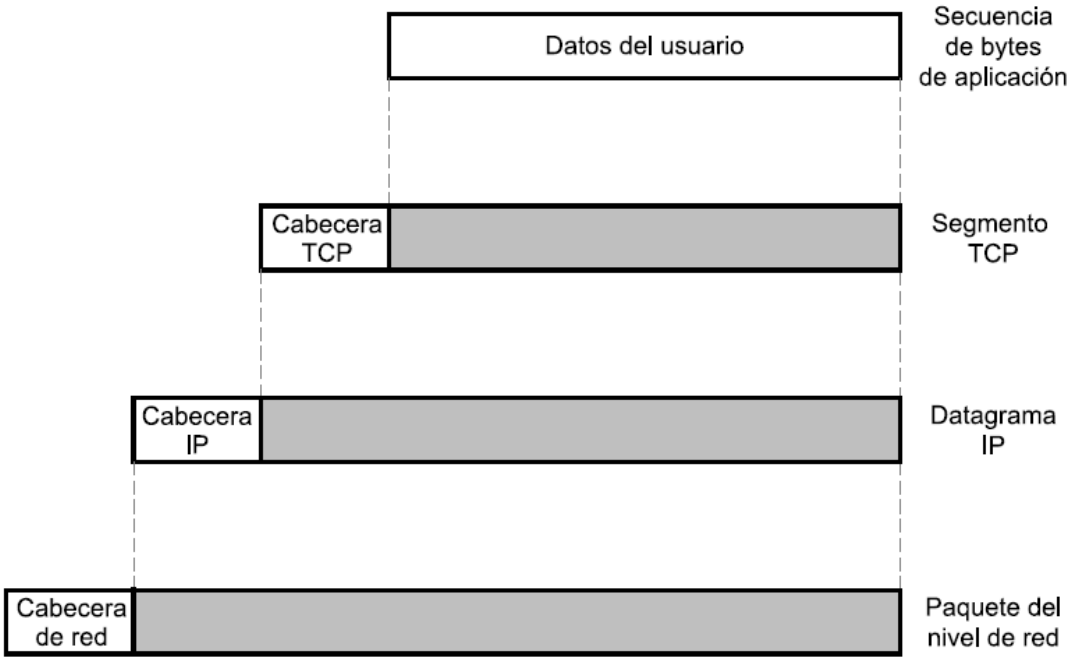


Figura 2.14. Unidades de datos de protocolo en la arquitectura TCP/IP.

Finalmente, cada datagrama IP se pasa a la capa de acceso a la red para que se envíe a través de la primera subred. La capa de acceso a la red añade su propia cabecera, creando un paquete, o trama. El paquete se transmite a través de la subred al dispositivo de encaminamiento J. La cabecera del paquete contiene la información que la subred necesita para transferir los datos. La cabecera puede contener, entre otros, los siguientes campos:

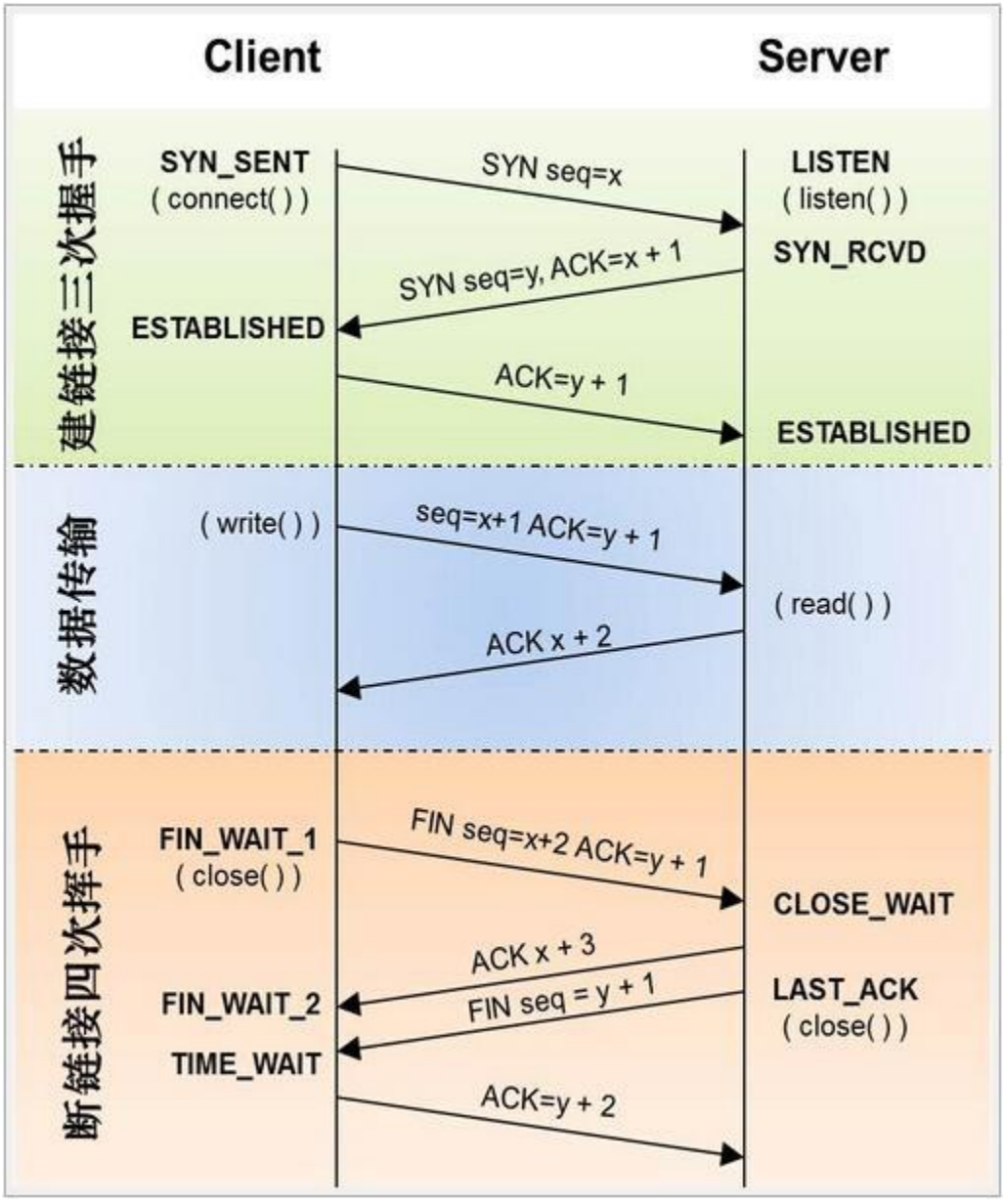
- **Dirección de la subred destino**: la subred debe conocer a qué dispositivo se debe entregar el paquete.
- **Funciones solicitadas**: el protocolo de acceso a la red puede solicitar la utilización de ciertas funciones ofrecidas por la subred, por ejemplo, la utilización de prioridades.

En el dispositivo de encaminamiento J, la cabecera del paquete se elimina y, posteriormente, se examina la cabecera IP. El módulo IP del dispositivo de encaminamiento direcciona el paquete a través de la subred 2 hacia B basándose en la dirección destino que contenga la cabecera IP. Para hacer esto, se le añade al datagrama una cabecera de acceso a la red.

Cuando se reciben los datos en B, ocurre el proceso inverso. En cada capa se elimina la cabecera correspondiente y el resto se pasa a la capa inmediatamente superior, hasta que los datos de usuario originales alcancen al proceso destino.

Como nota final, recuérdese que el nombre genérico del bloque de datos intercambiado en cualquier nivel se denomina unidad de datos del protocolo (PDU, Protocol Data Unit). Consecuentemente, el segmento TCP es la PDU del protocolo TCP.

Diagramas de Saludos de Tres y Cuatro Vías



Comparación de TCP Contra UDP

La mayor parte de aplicaciones que se ejecutan usando la arquitectura TCP/IP usan como protocolo de transporte TCP. TCP proporciona una conexión fiable para transferir los datos entre las aplicaciones.

Una conexión es simplemente una asociación lógica de carácter temporal entre dos entidades de sistemas distintos. Cada PDU de TCP, denominada segmento TCP, contiene en la cabecera la identificación de los puertos origen y destino, los cuales corresponden con los puntos de acceso al servicio (SAP) de la arquitectura OSI. Los valores de los puertos identifican a los respectivos usuarios (aplicaciones) de las dos entidades TCP.

Una conexión lógica alude a un par de puertos. Durante la conexión, cada entidad seguirá la pista de los segmentos TCP que vengan y vayan hacia la otra entidad, para así regular el flujo de segmentos y recuperar aquellos que se pierdan o dañen.

Además del protocolo TCP, la arquitectura TCP/IP usa otro protocolo de transporte: el protocolo de datagramas de usuario, UDP (User Datagram Protocol). UDP no garantiza la entrega, la conservación del orden secuencial, ni la protección frente duplicados.

UDP posibilita el envío de mensajes entre aplicaciones con la complejidad mínima. Algunas aplicaciones orientadas a transacciones usan UDP. Un ejemplo es SNMP (Simple Network Management Protocol), el protocolo normalizado para la gestión en las redes TCP/IP.

Debido a su carácter no orientado a conexión, UDP en realidad tiene poca tarea que hacer. Básicamente, su cometido es añadir a IP la capacidad de identificar los puertos.

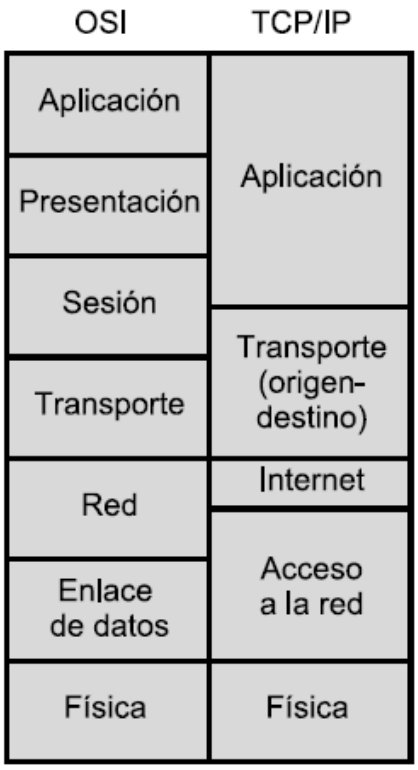
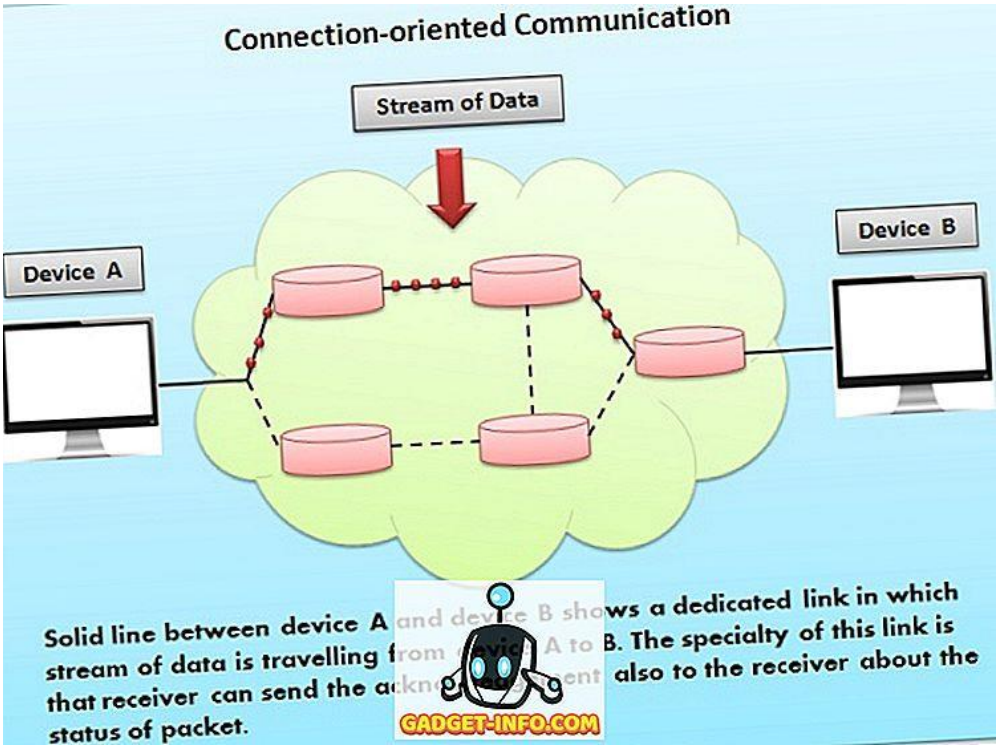
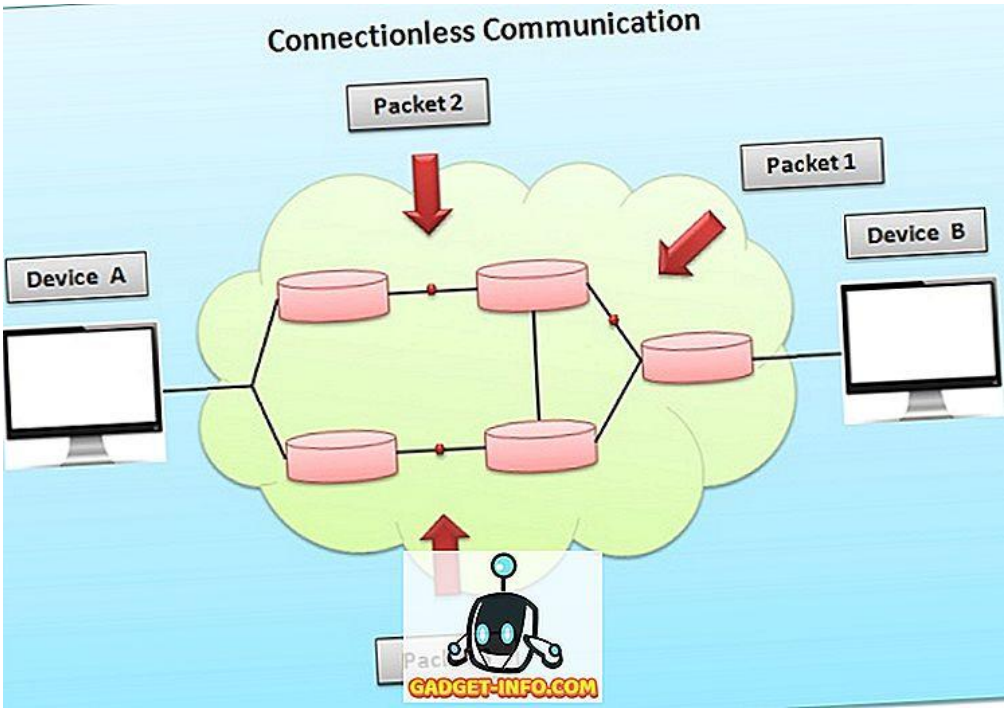


Figura 2.12. Comparación entre las arquitecturas de protocolos TCP/IP y OSI.

Diagramas de COS y CLOS





Ventana Deslizante

Técnicas de ventana deslizante Método de control de flujo en el que una estación que transmite puede enviar paquetes numerados dentro de un intervalo de números (ventana). La ventana cambia dinámicamente para permitir que se envíen paquetes adicionales.

TCP emplea una técnica de control de flujo basada en créditos que es, en cierta forma, diferente del control de flujo con ventana deslizante que encontramos en X.25 y HDLC. Básicamente, TCP separa las confirmaciones y la gestión del tamaño de la ventana deslizante.

Uso e Interpretación de las 9 Banderas de Control

1. **SYN: Synchronisation**
   En ese proceso se enviaba un SYN, es decir, se marcaba el bit de SYN en esa comunicación y era el establecimiento de la conexión. Luego el host de destino tenía que establecer también la conexión con otro SYN de vuelta, pero marcando también el ACK
2. **ACK: Acknowledgment**
   Este bit se marca para «agradecer» la recepción, en el proceso de desafío en 3 vías es lo que se envía para confirmar que hemos recibido el SYN inicial. Posteriormente se están enviando constantemente ACKs para confirmar la recepción de los segmentos enviados por el origen al destino.

3. **FIN: Finished**
   El flag FIN indica que ya no hay más datos desde el origen, así que este es el flag que se utilizará en el último segmento enviado por el origen.
4. **RST: Reset**
   Este flag se envía desde el destino al origen y se envía en el momento en el que el destino recibe un segmento que no se espera y que no debería de haber llegado.
5. **PSH: Push**
   El flag PSH indica al receptor que tiene que procesar los segmentos a medida que son recibidos y que no se deben de almacenar en un buffer.
   Hay que tener en cuenta que la capa de transporte, el protocolo TCP en nuestro caso, espera un tiempo antes de enviar el segmento a recibir suficientes datos de capas superiores y en el receptor pasa lo mismo pero al revés.
6. **URG: Urgent**
   El flag URG es algo parecido a PSH, pero en este caso lo que hacemos es priorizar aquellos segmentos marcados como urgente sobre los no marcados.
   Este flag ha tenido muchos problemas y de hecho desde el año 2011 se indica claramente que no debe de utilizarse más en la RFC6093, la cual me he estado revisando para preparar el audio, y en la página 7 indica claramente «Las nuevas aplicaciones NO DEBEN emplear el mecanismo urgente TCP«.
   Se decidió dejarlo de usar porque había ambigüedades en RFCs anteriores en cuanto a la semántica del mismo.
7. **ECE: Explicit Congestion Notification [ECN]-Echo**
   Este flag lo que hace es indicar que el peer de TCP permite ECN (Explicit Congestion Notification).
   En ECN a día de hoy está soportado por cualquier distribución de GNU/Linux y la idea de este mecanismo es evitar la pérdida de información en caso de congestión.
   Mientras se realiza el desafío en 3 vías una de las cosas que se hacen además del SYN, SYN/ACK, ACK es indicar que el nodo es compatible con ECN (Notificación de congestión explícita).
8. **CWR: Congestion Windows Reduced**
   El host emisor establece el flag CWR para indicar que recibió un segmento TCP con el flag ECE.
   Cuando se marca el flag CWR se reduce a la mitad el tamaño de la ventana en envío con el objetivo de ralentizar el envío de información.
9. **NS: Nonce Sum**

El flag NS es un flag experimental que se utiliza contra envíos maliciosos por parte del origen todavía están viendo cómo utilizarlo.

Socket

El concepto de socket, así como la programación de los sockets, fueron desarrollados en los años ochenta en el entorno de Unix como la interfaz de sockets de Berkeley.

Básicamente, un socket permite la comunicación entre un proceso cliente y un proceso servidor y puede ser orientado a conexión o no orientado a conexión. Un socket puede considerarse como un punto final en una comunicación.

Un socket cliente en una computadora utiliza una dirección para llamar a un socket servidor en otro computador. Una vez se han enlazado los sockets apropiados, los computadores pueden intercambiar datos.

Protocolos Pertenecientes a la Suite TCP/IP

El protocolo simple de transferencia de correo (SMTP, Simple Mail Transfer Protocol) proporciona una función básica de correo electrónico. Este protocolo establece un mecanismo para transferir mensajes entre computadores remotos.

Entre las características de SMTP cabe destacar la utilización de listas de mensajería, la gestión de acuses de recibo y el reenvío de mensajes. El protocolo SMTP no especifica cómo se crean los mensajes.

Para este fin se necesita un programa de correo electrónico nativo o un editor local. Una vez que se ha creado el mensaje, SMTP lo acepta y, utilizando TCP, lo envía al módulo SMTP del computador remoto.

En el receptor, el módulo SMTP utilizará su aplicación de correo electrónico local para almacenar el mensaje recibido en el buzón de correo del usuario destino.

El protocolo de transferencia de archivos (FTP, File Transfer Protocol) se utiliza para enviar archivos de un sistema a otro bajo el control del usuario. Se permite transmitir archivos tanto de texto como en binario. Además, el protocolo permite controlar el acceso de los usuarios.

Cuando un usuario solicita la transferencia de un archivo, FTP establece una conexión TCP con el sistema destino para intercambiar mensajes de control.

Esta conexión permite al usuario transmitir su identificador y contraseña, además de la identificación del archivo junto con las acciones a realizar sobre el mismo.

Una vez que el archivo se haya especificado y su transferencia haya sido aceptada, se establecerá una segunda conexión TCP a través de la cual se materializará la transferencia.

El archivo se transmite a través de la segunda conexión, sin necesidad de enviar información extra o cabeceras generadas por la capa de aplicación. Cuando la transferencia finaliza, se utiliza la conexión de control para indicar la finalización. Además, esta misma conexión estará disponible para aceptar nuevas órdenes de transferencia.

TELNET facilita la realización de conexiones remotas, mediante las cuales el usuario en un terminal o computador personal se conecta a un computador remoto y trabaja como si estuviera conectado directamente a ese computador. El protocolo se diseñó para trabajar con terminales poco sofisticados en modo scroll (avance de pantalla).

TELNET se implementa en dos módulos: el usuario TELNET interactúa con el módulo de E/S para comunicarse con un terminal local. Este convierte las particularidades de los terminales reales a una definición normalizada de terminal de red y viceversa.

El servidor TELNET interactúa con la aplicación, actuando como un sustituto del gestor del terminal, para que de esta forma el terminal remoto le parezca local a la aplicación. El tráfico entre el terminal del usuario y el servidor TELNET se lleva a cabo sobre una conexión TCP.

SEUDOCABECERA de TCP

El campo suma de comprobación se aplica a todo el segmento más una pseudocabecera incorporada en el momento del cálculo (tanto en la transmisión como en la recepción). La pseudocabecera incluye los siguientes campos de la cabecera IP: dirección red origen y destino, el protocolo y un campo de longitud del segmento. Con la inclusión de la pseudocabecera, TCP se protege ante un reparto erróneo de IP. Es decir, si IP entrega un segmento a una estación errónea, aunque el segmento esté libre de errores de bits, la entidad TCP receptora detectará el error de la entrega.

CONCLUSIÓN:

Tras elaborar el resumen ahora conocemos la metodología del TCP asi como sus ultimas actualizaciones, viendo asi que es un modelo vigente que se sigue usando hoy en día en el ámbito de la comunicación.

GLOSARIO:

- **Datagrama:**
  Un datagrama es una unidad de datos que se encuentra asociada a una red de conmutación de paquetes. Es habitual que estos se encuentren estructurados en secciones de cabecera y datos transmitidos en payload.

REFERENCIAS:

- STALLINGS, WILLIAM . (2004). COMUNICACIONES Y REDES DE COMPUTADORES. España: PEARSON EDUCACIÓN
- Eduardo Collado. (2020, May 6). Flags de TCP. https://www.eduardocollado.com/2020/03/13/flags-de-tcp/