Entendiendo TCP/IP

(Los fundamentos para comprender seguridad informática)

Toda la información que viaja por Internet (y en redes en general) está contenida en "frames" (paquetes). Estos paquetes están conformados por los "datos" que una cierta aplicación quiere enviar (por ejemplo envío: "dominio *yahoo.com* dame tu página web" al hacer *http://www.yahoo.com*). A éstos "datos" se le adicionan "headers" (encabezados) por los diferentes protocolos de TCP/IP. Lo malo, lo bueno, lo que quiere hacer mal, todo está dentro de los "frames".
TCP/IP es una suite (conjunto) de protocolos. Como ya dijimos, cada protocolo en un dado orden, agrega a los "datos" a ser enviados un "header". Allí se incluye la información del número IP del que envía, del que recibe, el puerto de la aplicación que envía los datos, el puerto destino y toda otra información relevante a la comunicación.
Quienes quieran realizar una maldad (hackers) o quienes desarrollen herramientas de protección (antivirus, firewalls, proxys) deberán conocer a fondo el detalle de cómo están conformados los paquetes.
En este artículo se explicará ese detalle. Se verá una introducción histórica de TCP/IP, el modelo que lo describe, y un análisis de los headers que se agregan cuando se conforman los paquetes.
Si desea entender cualquier artículo sobre seguridad o implementar alguna herramienta será indispensable tener claro lo aquí expuesto. En NEX 7 se explicitará un detalle ampliado sobre los protocolos IP, TCP y UDP.

TCP/IP

Los protocolos TCP/IP (también llamados "Internet Protocols") son la "amalgama" que conecta hoy la mayoría de las redes de computadoras. También son responsables de la existencia de Internet: la red de redes que nos permiten entre otras cosas enviar correo electrónico, poder ver páginas Web y realizar transacciones comerciales en materia de segundos sin tener un límite geográfico. Los protocolos TCP/IP fueron originalmente desarrollados para soportar tareas de investigación pero han logrado un alto grado de maduración y aceptación casi universal. Las investigaciones realizadas por el mundo académico fue financiado en su mayor parte con subsidios de las fuerzas armadas americanas, a través del proyecto ARPANET (Advanced Research Project for Networking. Año 1969).
En 1983 se divide en dos redes MILNET (de uso militar) e INTERNET (de uso académico). En 1990 INTERNET se hace comercial y surge el boom del e-commerce e infinidad de otros mundos.
TCP/IP se refiere a un conjunto (suite) de protocolos para comunicación de datos. La suite toma su nombre de dos de los protocolos que lo conforman: Transmition Control Protocol (TCP) e Internet Protocol (IP). La figura 1 nos detalla algunos de los protocolos más comunes que conforman la suite.

Modelos para describir la arquitectura de comunicación de datos

Un modelo arquitectónico fue desarrollado por la International Standards Organization (ISO) y usado para describir la estructura y función de los protocolos de comunicación de datos: OSI (Open Systems Interconnect Referente Model). Ver Figura 2.
Contiene siete capas (layers) que definen las funciones de los protocolos de comunicación de datos. TCP/IP puede ser descrito con el modelo OSI pero existe un modelo de arquitectura (alternativo) propio (ver figura 2, TCP/IP implementación) compuesto por cuatro capas.
Cada capa representa una función que se realiza en la transferencia de datos entre aplicaciones a través de la red. Se lo llama un "apilamiento" o "stack".
Una capa no define un solo protocolo. Define una función que puede ser realizada por un número de protocolos. Por ejemplo, un protocolo de transferencia de archivos (FTP) y una de correo electrónico (SMTP) proveen servicios al usuario y son parte del Application layer.
Cuando dos máquinas se comunican, cada protocolo se comunica con su "peer" (par). Un par es una implementación del mismo protocolo en la capa equivalente en el sistema remoto.
En principio cada protocolo debería solo interesarse de la comunicación con su peer. Sin embargo, deberá también haber un acuerdo de cómo pasar los datos entre capas dentro de una sola computadora. Los datos son pasados bajando por el "stack" de una capa a la otra hasta que es transmitida por los protocolos de la llamada "Physical Layer" por la red. Por otro lado los datos son tomados de la red y subidos a través del "stack" hasta la aplicación receptora.
Las capas individuales no necesitan saber como funcionan la capa superior e inferior a ella: solo como pasar los datos. (ver fig. 3)

Este aislamiento de funciones en cada capa minimiza el impacto sobre toda la suite, que se pueden producir por los avances tecnológicos.En cada capa del "stack" se adiciona información de control llamado "header" (encabezado) ya que se coloca al frente de los datos a transmitir (ver fig. 4)

Cada capa trata toda la información que recibe de las capas superiores como "datos" y adiciona "su" propio "header" (proceso llamado encapsulación). Cuando se recibe información sucede lo opuesto.
Es importante resaltar que cada capa define una estructura de datos independiente de las otras y su propia terminología que la describe. La figura 5 muestra los términos usados en las diferentes capas para referirse a los datos transmitidos (i.e un "datagram" tiene el "header" correspondiente a la internet layer y lo que le pasa la capa superior).

Descripción de cada layer (capa)

Las figuras 6,7 y 8 muestran una representación pictórica de la estructura de los "headers" y datos. Los "headers" están conformados por varios "words" de 32 bits donde se incluye información. Recordar que cada layer tiene su propia estructura (Fig. 4.) y agrega un "header" a lo que recibe de la capa superior que lo toma como "datos". Esta información adicional que garantiza el "delivery"(entrega) se llama "encapsulación". Cuando se reciben "datos" lo opuesto sucede.

TCP/IP Suite — Figura nro. 1

Los protocolos asociados con TCP/IP incluyen los siguientes:

IP	Internet Protocol
TCP	Transmission Control Protocol
IGMP	Internet Group Management Protocol
ICMP	Internet Control Message Protocol
ARP	Address Resolution Protocol
RARP	Reverse Address Resolution Protocol
UDP	User Datagram Protocol
FTP	File Transfer Protocol
SMTP	Simple Mail Transfer Protocol
SNMP	Simple Network Management Protocol

Arquitectura TCP/IP — Figura nro. 2

Modelo OSI	Implementación TCP/IP
Application Layer	Application Layer
Presentation Layer	
Session Layer	
Transport Layer	Transport Layer
Network Layer	Internet Layer
Data Link Layer	Network Interface Layer
Physical Layer	

Cada layer elimina su "header" antes de pasar los "datos" a la capa superior. Cuando la información sube el stack, lo que llega de la capa inferior es interpretada como header y datos.
La información de los estándares de los diferentes protocolos es desarrollada y publicada a través de los llamados "Request For Comments". (Ver nota)

NETWORK ACCESS LAYER (Capa de Acceso a la red)

La Network Acess Layer es la de más abajo en la jerarquía de protocolos TCP/IP. Los protocolos en está capa proveen el modo en que el sistema envía los datos a otros dispositivos en una red a la que esta directamente conectado.
Si aparecen nuevas tecnologías de hardware deberán desarrollarse nuevos protocolos para la Network Access Layer. Hay muchos protocolos de acceso: uno para cada Standard de red física. (Ethernet, Token Ring, Cobre-teléfono, Fibra.)
Las funciones que se realizan a este nivel incluyen encapsulación de datagramas IP: ("frames" que se transmiten por la red) y el mapeo de números IP a las direcciones físicas usadas por la red (i.e el MAC address)

Figura nro. 3

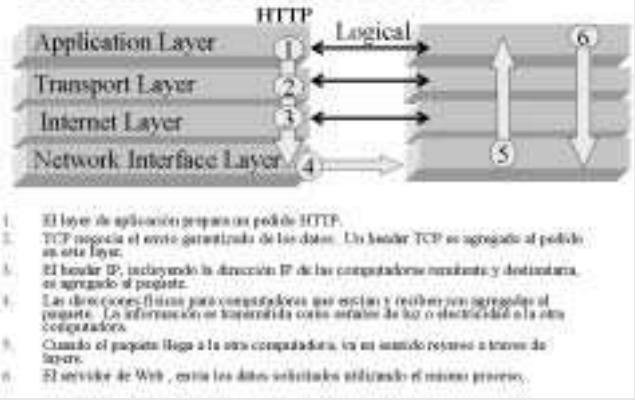
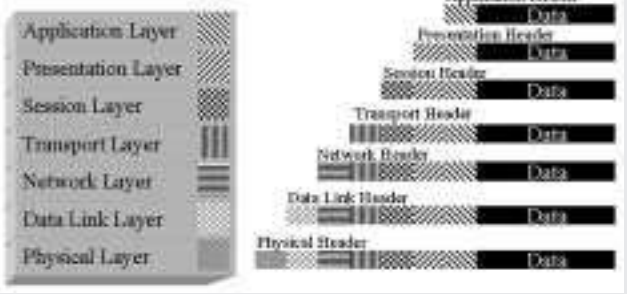


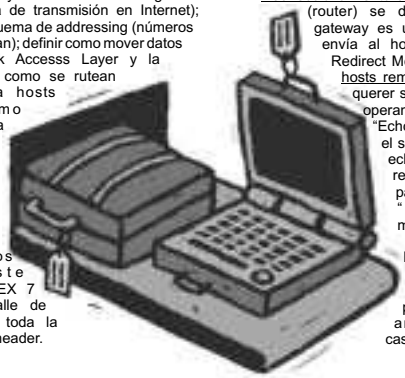
Figura nro. 4



Dos ejemplos de RFCs que definen protocolos de esta capa son:
RFC 826 ARP (Address Resolution Protocol) resuelve numeros IP a MAC addressses.
RFC 894 especifica como se encapsulan los datagramas para transmitirlos por las redes Ethernet.

Internet Layer.
Esta es la capa arriba de la Network Access Layer. El "Internet Protocol"(IP) es el corazón de TCP/IP y el protocolo más importante de esta layer. Todos los protocolos en capas superiores e inferiores lo usan para "el delivery" de datos. IP está complementado por ICMP (Internet Control Message Protocol).

IP (Internet Protocol)
IP es el protocolo sobre el que se basa Internet. IP es un protocolo connectionless. (Ver nota). Además se basa en protocolos de otras layers para realizar "error detection y recovery".
Sus funciones incluyen: definición de "datagrama" (la unidad básica de transmisión en Internet); definición del esquema de addressing (números IP y como funcionan); definir como mover datos entre la Network Accesss Layer y la Transport Layer; como se rutean "datagramas" a hosts remotos; como realiza fragmentacion y re-armado de "datagramas".
La figura 6 nos muestra un esquema del datagrama IP.
Recomendamos estudiar este esquema. En NEX 7 veremos el detalle de cómo se utiliza toda la información en el header.



ICMP (Internet Control Message Protocol). Protocolo de Control de Mensajes de Internet.
Es complementario al Internet Protocol y fue definido por el RFC 792. Forma parte de la Internet Layer. Manda mensajes realizando tareas de control como reporte de errores e información de funcionamiento de TCP/IP

Algunos ejemplos de sus funcionalidades: Control de flujo: Si los datagramas llegan muy rápido para ser procesados, el host que los recibe o un gateway (router) en el camino, manda el llamado ICMP "Source Quench Message" a quien envió el mensaje. Este detiene temporariamente los envíos. Destinos no accesibles:(Unreachable). Si un sistema se da cuenta que el destino de un paquete es no accesible envía a la fuente (source) un "Destination Unreachable Message". Si el destino no accesible es un host o network, el mensaje lo envía un gateway (router) intermedio. Pero si el destino es un "puerto" no accesible, el host destino envía el mensaje. Redireccionamiento de ruta: Si un gateway (router) se da cuenta que otro gateway es una mejor opción, le envía al host fuente un "ICMP Redirect Message". Chequeo de hosts remotos. Un host puede querer saber si otro host está operando. Envía un ICMP "Echo Message". Cuando el segundo host recibe el echo message, contesta re- enviando el mismo paquete. El comando "ping" usa este mensaje.

La Tabla 1 muestra los códigos que son utilizados por ICMP para los ejemplos anteriores y otros casos.

Tabla nro. 1

Tipo de código	Mensaje ICMP
0	Respuesta a eco (respuesta a PING)
3	Destino inaccesible
4	Source query
5	Redirección
8	Eco (petición de PING)
11	Tiempo de vida excedido (TTL)
12	Problema en algún parámetro
13	Petición de marca de tiempo
14	Respuesta de marca de tiempo
17	Petición de máscara de red
18	Respuesta de máscara de red

Figura nro. 5

Layer	TCP	UDP
Application Layer	Stream	Message
Transport Layer	Segment	Packet
Internet Layer	Datagram	Datagram
Network Access Layer	Frame	frame

Protocolos

Cuando las computadoras se comunican, es necesario definir un conjunto de reglas que gobiernen su comunicación. Este conjunto de reglas se llaman protocolos.
Los protocolos TCP/IP están disponibles para cualquiera, desarrollados y cambiados por consenso. Y, han sido adoptados universalmente, lo que permite la conectividad de redes heterogéneas.

TRANSPORT LAYER

Los dos protocolos más importantes en esta capa son TCP (Transmission Control Protocol) y el UDP (User Datagram Protocol). TCP nos provee un servicio de entrega de datos confiable. Incluye detección y corrección de errores end-to-end (de punta a punta). UDP provee un servicio de entrega "connectionless" y mucho más reducido. Ambos, además, mueven los datos entre los Application layer e Internet layer dentro de la misma máquina. Quien programe una aplicación dada elegirá qué servicio es el más apropiado.

UDP (User Datagram Protocol)
UDP es un protocolo "connectionless" y no-confiable (no-confiable significando que no existe dentro del protocolo una infraestructura que cerifique que los datos llegan al destino correctamente). El header UDP (ver figura 7) utiliza en la "word1" 16 bits para detallar el Source-Port (puerto fuente) y otros 16 para el Destination-Port (puerto destino). De este modo sabe (por el número de puerto) qué aplicación lo envió y cuál lo recibirá.

¿Porqué decide quien programa una aplicación usar UDP?. Puede haber varias razones. Por ejemplo, si la cantidad de datos es muy pequeña, el overhead de crear la conexión y asegurarse la entrega puede ser mayor que re-transmitir los datos. Aplicaciones del tipo pregunta-respuesta son excelentes candidatos.

La respuesta misma se puede usar como un aviso positivo de entrega. Si no llega una respuesta en un dado tiempo la aplicación vuelve a enviar su pedido. Puede también ser que una dada aplicación provea su propia infraestructura para entrega confiable y no necesitara una infraestructura más compleja que UDP.
Ver Fig. 7

TCP (Transmission Control Protocol) (Procolo de Control de Transmisión)
Las aplicaciones que necesiten que se les provea de una infraestructura confiable usarán TCP. Usando TCP estará segura de que los datos llegaron a destino y en la secuencia adecuada. TCP es un protocolo confiable, "connection-oriented" y "byte-stream".

En NEX 7 ampliaremos detalles de TCP. Un estudio de la figura 7 nos indica qué información utiliza para establecer lo que se llama el "three way handshake" (estrechado de mano de tres pasos). En el word1 (al igual que en UDP) se envia la información de los puertos origen y destino. Pero en este caso es enviada mucha más información.

APPLICATION LAYER

PROTOCOLOS DE CAPA DE APLICACIÓN
En la capa superior de la arquitectura TCP/IP está la Application Layer. Esta incluye todos los procesos que utilizan a la Transport Layer como medio de entrega de datos.
Es la parte de TCP/IP donde se procesan los pedidos de "datos" o servicios. Las aplicaciones de esta capa están también esperando pedidos para procesar y están "escuchando" por sus puertos respectivos.

La Application Layer NO es donde está corriendo un procesador de palabras (por ejemplo WORD), una hoja de cálculo o un browser de Internet (Netscape o Internet Explorer). Las aplicaciones que corren en esta capa, SI, interactúan con los procesadores de texto, programas de hoja de cálculo y otras.

Figura nro. 6

Versión IP (4 bits)	Longitud de la cabecera (4 bits)	Tipo de servicio (8 bits)	Longitud total (16 bits)
Identificación (ID) del fragmento (16 bits)		R / D F / M F	Posición de fragmento (13 bits)
Tiempo de vida (TTL) (8 bits)	Protocolo (8 bits)		Suma de comprobación de la cabecera (16 bits)
Dirección IP de origen (32 bits)			
Dirección IP de destino (32 bits)			
Opciones (longitud variable que se completa con ceros hasta 40 bits)			
Datos			

Figura nro. 7

Número de puerto de origen (16 bits)	Número de puerto de destino (16 bits)
Longitud (16 bits)	Suma de comprobación (16 bits)
Datos	

Figura nro. 8

Número de puerto de origen (16 bits)								Número de puerto de destino (16 bits)
Número de secuencia (32 bits)								
Número de acuse de recibo (32 bits)								
Desplazamiento (4 bits)	Reservado (6 bits)	U	A	P	R	S	F	Ventana (16 bits)
Suma de comprobación (16 bits)								Puntero urgente (16 bits)
Opciones (Longitud variable y relleno con ceros)								
Datos								

Figura nro. 9

Combinación de indicadores	Significado
SYN	Primer paquete de la conexión que especifica el pedido de comunicación con el equipo destino
SYN/ACK	El segundo equipo responde y envía su SYN
ACK	En cada envío se activa este bit para asegurar que el envío anterior se ha recibido correctamente
FIN	Señal enviada por el equipo que está preparado para cerrar la conexión
FIN/ACK	Señal enviada por el segundo equipo para aceptar el cierre de conexión y validar el estado de recepción de paquetes
RST	El paquete RST se envía para dar aviso de recepción de paquetes no esperados. Un caso claro es el de un paquete SYN/ACK que llega sin haber recibido previamente un paquete SYN

Los protocolos SMTP, http, Telnet, POP, DNS o FTP son ejemplos de protocolos de esta layer.

Request For Comment (RFC).
La naturaleza abierta de los protocolos TCP/IP requiere documentación pública de los estándares. La mayor parte de la información de TCP/IP se publica como Request for Comments (RFC). Como implica el nombre, el estilo y contenido de estos documentos es poco rígido. Los RFC contienen información bastante completa y no se remiten solamente a las especificaciones formales.

Protocolos Connection oriented y Protocolos connectionless (no orientado a connección)
Protocolo connection oriented: intercambia información de control con el sistema remoto (llamada handshake (dado de mano), para verificar que está listo para recibir datos antes de enviarlos.
Se establece una "connection" end-to-end.

(Ejemplos TCP)
Protocolo connectionless: Que NO intercambia información de control.

¿Porque triunfó TCP/IP sobre otras alternativas?
Son protocolos abiertos, disponibles gratuitamente y desarrollados en forma independiente de cualquier vendor de hardware o sistema operativo. Son independientes de cualquier hardware físico particular. TCP/IP puede correr sobre Ethernet, Token Ring, línea telefónica dial-up, X.25 net y virtualmente cualquier otro tipo de medio físico de transmisión. Un esquema de "addressing" (direccionamiento) universal que permite a cualquier dispositivo TCP/IP dirigirse en forma única a cualquier otro dispositivo de la red aún cuando la red sea tan grande como el world-wide Internet.