Video: Protocolo TCP de enlace de tres vías (7 min)

Tengo capturas de pantalla de una captura de paquetes de Wireshark con el proceso de enlace de 3 vías de TCP y la finalización de una conversación de TCP. Veamos las capturas de pantalla para tener una idea de cómo funciona.

TCP es un protocolo orientado a la conexión, o sea, una conexión completa se debe establecer primero para que los datos se puedan enviar o recibir. El enlace de 3 vías TCP inicia la conexión. Cuando la conexión debe finalizar, por ejemplo, una conexión con un servidor web, y cierra el navegador web, la conexión se termina con un enlace de dos vías.

Un enlace de tres vías TCP consta de 3 pasos, [SYN], [SYN, ACK] y [ACK]. SYN significa sincronización; ACK significa reconocimiento. Primero, el host de origen envía un segmento de sincronización. Luego, el host de destino envía un reconocimiento y su propio segmento de sincronización. Luego el host inicial envía un segmento de reconocimiento; por lo tanto, [SYN], [SYN, ACK] y [ACK]. Podemos verlo aquí, en la captura de pantalla. Si miramos la ventana Packet List, en Packets 10, 11 y 12, vemos un [SYN], un [SYN, ACK], y un [ACK]. Es el protocolo de enlace de 3 vías. Si vemos el paquete inicial del enlace de tres vías, el segmento de [SYN] de esta parte, podemos ver que el número de secuencia es 0. El comienzo del enlace de tres vías es el número 0 porque es el primer paquete de la conexión o conversación entre dos hosts, o en este caso, los hosts del servidor. El número de secuencia es un número aleatorio de 32 bits denominado ISN o número de secuencia inicial. Este número (o ISN) se selecciona aleatoriamente al principio de cada conversación TCP. Esto ayuda a proteger contra ataques a la conexión TCP. Wireshark toma el número aleatorio de 32 bits y lo convierte en 0. Luego incrementa los números de secuencia y reconocimiento a partir de ahí. Esto facilita la lectura y sigue los segmentos en orden con el programa Wireshark.

Analicemos algunos de los detalles de este segmento inicial [SYN]. Vamos a la ventana Packet Details y podemos ver Sequence number: 0, que es un número de secuencia relativa. Si observamos Flags, vemos que se fijó Syn bit. Puede verlo aquí con un 1. En el próximo paquete, número 11, el servidor responde al segmento de sincronización inicial. Vamos a la próxima pantalla; ahora el paquete 11 está resaltado. El servidor responde con un reconocimiento cuyo número es 0 y se envía el reconocimiento 1, y el número de secuencia inicial, con el número de secuencia relativa 0, se ha incrementado y se ha enviado el reconocimiento 1. Podemos ver en la ventana Protocol Details, el número de reconocimiento 1. Es el número de reconocimiento relativo. El servidor también envió su segmento de sincronización, que es 0, ya que es la conversación inicial en otra dirección. Si miramos la ventana Details, podemos ver que el número de secuencia es 0, y que es el número de secuencia relativa del servidor al host. Si observamos Flags, tanto SYN como ACK se configuraron.

Si vemos la captura siguiente, en el paquete 12, paso 3, del enlace de 3 vías, host 10.1.1.1 responde con un reconocimiento o [ACK], y si analizamos la ventana Protocol Details, vemos que el reconocimiento es 1, incrementa el segmento de sincronización del servidor en 1. Puede verse que se fijó el bit de reconocimiento, pero vea que el bit de sincronización no se fijó. Es el final del protocolo de enlace de 3 vías.

Veamos cómo se interrumpirá la conexión TCP. En la próxima pantalla, se puede ver que en el paquete 16, el servidor se comunica con el host en 10.1.1.1, y envió un reconocimiento con un final o FIN, y un reconocimiento o ACK. En este segmento, tenemos [FIN, ACK]. El FIN finaliza la conversación. Se fijó la señal de reconocimiento ya que el enlace de 3 vías primero se fijó, y en cada segmento enviado luego, se fija señal de reconocimiento. Puede ver en el paquete número 17, el host respondió al servidor con reconocimiento de que la conversación finalizó. Es un enlace de dos vías. Un [FIN, ACK] y un [ACK]. Si nos anticipamos, en la ventana Packet List, el paquete 18, se puede ver que el host 10.1.1.1 envía al servidor su propio FIN y reconocimiento, y luego el servidor responde con su [ACK]. Tenemos dos enlaces de dos vías para finalizar la conexión. Si vuelve a la captura anterior, y observa Protocol Details o Packet Details, se puede ver aquí el segmento TCP, los indicadores, observe el 1 del reconocimiento y luego el 1 del final o indicador de final. Aquí se configuró. Observe que los reconocimientos subieron hasta 374, lo que indica que estas capturas se generaron de dos capturas de paquetes separadas en Wireshark. Puede ver en estas dos capturas más recientes cómo termina la conversación con 2 enlaces de dos vías, un [FIN, ACK] y un [ACK], y luego otro que va en otra dirección.