Vidéo - Connexion TCP en trois étapes (7 min)

Voici des captures d'écran de paquets Wireshark qui illustrent le processus de connexion TCP en trois étapes et l'arrêt d'une conversation TCP. Analysons ces captures d'écran pour comprendre comment ça marche.

Le protocole TCP est un protocole orienté connexion, c'est-à-dire qu'une connexion de bout en bout doit d'abord être établie pour que les données puissent être envoyées ou reçues. La connexion TCP en trois étapes est la première phase. Lorsque la connexion doit être interrompue, par exemple, lorsque vous êtes connecté à un serveur Web et que vous fermez le navigateur Web, la déconnexion se fait en deux échanges bidirectionnels.

Une connexion TCP comprend donc trois étapes : [SYN], [SYN, ACK] et [ACK]. SYN signifie Synchronisation et ACK signifie Accusé de réception. L'hôte source envoie un segment de synchronisation, l'hôte de destination envoie un accusé de réception ainsi que son propre segment de synchronisation, puis l'hôte source envoie un segment d'accusé de réception, donc [SYN], [SYN, ACK] et [ACK]. C'est ce que nous montre cette capture d'écran. Si nous examinons la liste de paquets, en particulier les paquets 10, 11, et 12, nous avons un [SYN], un [SYN, ACK] et un [ACK]. Il s'agit de la connexion en trois étapes. Si nous examinons le paquet initial de la connexion en trois étapes, à savoir ce segment [SYN], en haut, nous voyons que le numéro d'ordre est 0. Le premier numéro d'ordre d'une connexion en trois étapes est 0, car il s'agit du premier paquet de la connexion ou de la conversation entre deux hôtes, ou dans ce cas, entre un hôte et un serveur. Le numéro d'ordre est en fait un numéro aléatoire de 32 bits appelé l'ISN ou Numéro d'ordre initial. Ce numéro, ou ISN, est sélectionné aléatoirement au début de chaque conversation TCP. Cela permet de protéger les connexions TCP des attaques. Wireshark prend ce numéro aléatoire de 32 bits et le convertit en 0. Il incrémente ensuite les numéros d'ordre et les accusés de réception à partir de là. Cela facilite la lecture et le suivi des segments dans le programme Wireshark.

Examinons un peu plus en détail ce premier segment [SYN]. Regardons maintenant dans la fenêtre de détail du paquet. Nous voyons le numéro d'ordre 0, qui est le numéro d'ordre relatif. En examinant les indicateurs, nous constatons que le bit Syn a été défini. Il est marqué d'un 1. Dans le paquet suivant, le n° 11, le serveur répond au segment de synchronisation initial. Passons à la capture d'écran suivante, dans laquelle le paquet 11 est surligné. Le serveur répond avec un accusé de réception, confirmant la réception du numéro d'ordre 0, puis envoie l'accusé de réception 1. Le numéro d'ordre initial, avec le numéro d'ordre relatif 0, a donc été incrémenté et l'accusé de réception 1 a été envoyé. Dans la fenêtre détaillée du protocole, nous voyons le numéro de réception 1 qui est un numéro relatif. Le serveur a également envoyé son propre segment SYN, dont le numéro est 0 puisqu'il s'agit de la 1ère conversation dans l'autre sens. Dans la fenêtre détaillée, nous voyons que le numéro d'ordre est 0 ; il s'agit du numéro d'ordre relatif du serveur vers l'hôte. D'après les indicateurs, les bits SYN et ACK ont tous deux été définis.

Dans la capture d'écran suivante, dans le paquet 12, qui correspond à la dernière des 3 étapes de la connexion, l'hôte 10.1.1.1 répond avec un accusé de réception, ou [ACK], et si nous regardons dans la fenêtre détaillée du protocole, nous constatons que le numéro de réception est 1, ce qui incrémente le segment de synchronisation du serveur de 1. Vous remarquerez que le bit d'accusé de réception est défini, ce qui n'est pas le cas du bit Syn. C'est la dernière phase de la connexion en trois étapes.

Voyons comment la connexion TCP se termine. Passons à la capture d'écran suivante. Dans le paquet 16, le serveur communique avec l'hôte à l'adresse 10.1.1.1 et a envoyé un segment de fin, ou FIN, ainsi qu'un accusé de réception, ou ACK. Dans ce segment, nous avons donc [FIN, ACK]. FIN met fin à la conversation. L'indicateur d'accusé de réception est défini, car la connexion en trois étapes a d'abord été établie et pour chaque segment envoyé par la suite, l'indicateur d'accusé de réception a été défini (Set). Comme vous le constatez dans le paquet suivant, n° 17, l'hôte a répondu au serveur en accusant réception de la fin de la conversation. Il s'agit d'un échange en deux étapes. Un [FIN, ACK] et un [ACK]. Si nous examinons le paquet 18 dans la liste de paquets, nous voyons que l'hôte 10.1.1.1 envoie ensuite son propre FIN et accusé de réception, et que le serveur répond avec son propre [ACK]. Il y a donc deux échanges bidirectionnels pour mettre fin à la connexion. Si nous revenons à la capture d'écran précédente, et examinons les données détaillées du protocole ou du paquet, nous voyons les indicateurs du segment TCP. Remarquez le 1 pour l'accusé de réception, puis le 1 pour la fin de la conversation dont le statut a été défini. Notez que les numéros d'accusés de réception vont jusqu'à 374, ce qui laisse à penser que ces captures d'écran proviennent de deux captures de paquets distinctes dans Wireshark. Dans ces deux dernières captures d'écran, nous voyons la fin de la conversation avec deux échanges bidirectionnels, [FIN, ACK] et [ACK], puis un autre dans l'autre sens.