视频 - TCP 三次握手（7 分钟）

我有一些 Wireshark 数据包捕获的屏幕截图展示了一次 TCP 三次握手的过程和 TCP 会话的终止。让我们分析一下这些屏幕截图，以了解它的工作原理。

TCP 是一个面向连接的协议，这意味着在发送或接收数据之前需要首先建立一个端到端连接。TCP 三次握手发起该连接。当连接最后需要终止时，例如，假设它与 Web 服务器有连接，而且您关闭了 Web 浏览器，该连接会通过两次二次握手来终止。

TCP 三次握手包括三个步骤，一次 [SYN]，一次 [SYN， ACK] 和一次 [ACK]]。SYN 表示同步，ACK 表示确认。首先，发起连线的主机发送一个同步数据段，响应的主机发送一条确认信息和它自己的同步数据段，然后初始主机发送一个确认数据段，这就是 [SYN]、[SYN，ACK] 和 [ACK]]。可以在此屏幕截图的顶部看到此过程。如果查看数据包列表窗口，在数据包 10、11 和 12 上，可以看到一个 [SYN]、一个 [SYN，ACK]和一个 [ACK]]。这就是三次握手。如果查看三次握手中的初始数据包，[SYN] 数据段位于顶部，可以看到序列号为 0。在三次握手的开头序列号为 0，因为它是两个主机或者在本例中为服务器中的主机之间的连接或对话中的第一个数据包。该序列号实际上是一个 32 位随机数，称为 ISN 或初始序列号。这个随机数或 ISN 是在每个 TCP 对话的开头随机选择的。这有助于防御 TCP 连接劫持攻击。Wireshark 获取这个 32 位随机数并将其转换为 0。它然后递增该序列号和确认号。这样更易于使用 Wireshark 程序按顺序读取和理解这些数据段。

我们看看这个初始[SYN] 数据段的一些细节。我们进入数据包详细信息窗口，可以看到序列号：0，它是一个（相对序列号）。如果查看标志，可以看到 Syn 位已设置。可以看到这里有一个 1。在下一个数据包中，数据包编号为 11，服务器响应初始同步数据段。我转到下一个屏幕截图，现在数据包 11 已高亮显示。服务器使用一个确认序列号 0 和发送确认号 1 作为响应，所以初始序列号（相对序列号）0 已递增，发送了确认号 1。可以在协议详细信息窗口中看到确认号：1，这是相对确认号。服务器还发送了自己的同步数据段，而且这个编号是 0，因为它是反向传输的初始对话。如果查看详细信息窗口，可以看到序列号为 0，这是一个从服务器传输到主机的相对序列号。查看这里的标志，SYN 和 ACK 位都已设置。

转到下一个屏幕截图，在数据包 12 中，也就是三次握手中的第 3 步，主机 10.1.1.1 使用一条确认信息或 [ACK] 作为响应，如果查看协议详细信息窗口，可以看到确认号是 1，服务器同步数据段递增了 1。在这里可看到确认位已设置，但请注意 Syn 位未设置。这是三次握手中的最后一个阶段。

我们看看 TCP 连接如何终止。我将转到下一个屏幕截图，可以看到，在数据包 16 中，服务器正在与地址为 10.1.1.1 的主机通信，并发送了一个数据段，其中包含一个 Finish 或 FIN 和一条确认信息或 ACK。在这个数据段中，有一个 [FIN，ACK]。FIN 结束会话。确认标志已设置，因为三次握手已首次建立，而且在随后发送的每个数据段中，确认标志都已设置。可以看到，在下一个数据包（数据包 17）中，主机使用一条确认信息回复了服务器，确认会话已结束。这是一次二次握手。一个 [FIN，ACK] 和一个 [ACK]]。如果在数据包列表窗口中继续查看数据包 18，可以看到主机 10.1.1.1 接着向服务器发送它自己的 FIN 和确认信息，然后服务器以自己的 [ACK] 作为回复。所以我们通过两次二次握手来终止连接。如果返回到上一个屏幕截图，查看协议详细信息或数据包详细信息，可以在 TCP 数据段中看到这些标志，可以注意到确认和 FIN 标志中都已设置了 1。可以注意到确认信息数高达 374 条，表明这些屏幕截图可能是从 Wireshark 中的两次不同的数据包捕获中生成的。可以在最后两个屏幕截图中看到如何通过两次二次握手，一个 [FIN，ACK] 和一个 [ACK]，来结束对话，然后反向建立另一个会话。另一方向也是如此。