8.1.3 TCP 的三次握手

所有基于 TCP 的通信都需要从两台主机的握手开始。这个握手过程主要希望能达到这样一些目的。

- 保证传输主机可以确定目的主机在线并且进行通信。
- 让传输主机确定目标主机在监听传输主机试图连接的端口。
- 允许传输主机向目标主机发送它的起始序列号，使得两台主机可以将这一会话保持得井然有序。

TCP 握手分为 3 个步骤，如图 8-5 所示。在第一步中，主动发起通信的设备（主机 A）向目标（主机 B）发送了一个 TCP 数据包。这个初始数据包除了底层协议头之外不包含任何数据。这个数据包的 TCP 头设置了 SYN 标志，并包含了在通信过程中会用到的初始序列号和最大分段大小（MSS）。主机 B 对于这个数据包回复了一个类似的设置了 SYN 和 ACK 标志以及包含了它初始序列号的数据包。最后，主机 A 向主机 B 发送最后一个仅设置了 ACK 标志的数据包。在这个过程完成之后，双方设备应该已经具有了开始正常通信所需的信息。

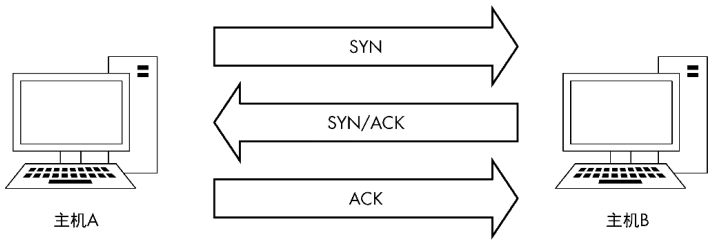

图 8-5　TCP 三次握手

> 注意
>
> TCP 数据包在称呼上通常会被其设置的标志所代表。比如，对于设置了 SYN 标志的 TCP 数据包，我们将会简称其为 SYN 包。因此 TCP 握手过程中使用的数据包会被称为 SYN 包、SYN/ACK 包和 ACK 包。

打开 tcp_handshake.pcapng，可以更直观地看到这个过程。Wireshark 为了分析的简便，引入了一个特性，可以将 TCP 数据包的序列号替换为相对值。但在这里，我们将这个功能关闭，以便于能看到实际的序列号值。选择 Edit -> Preferences，展开 Protocols 并选择 TCP，然后取消勾选 Relative Sequence Numbers and Window Scaling 框，并单击 OK 就可以禁用了。

这个捕获中的第一个数据包是我们的初始 SYN 数据包（见图 8-6）。这个数据包从 172.16.16.128 的 2826 端口发往 212.58.226.142 的 80 端口。我们可以看到这里传输的序列号是 3691127924。

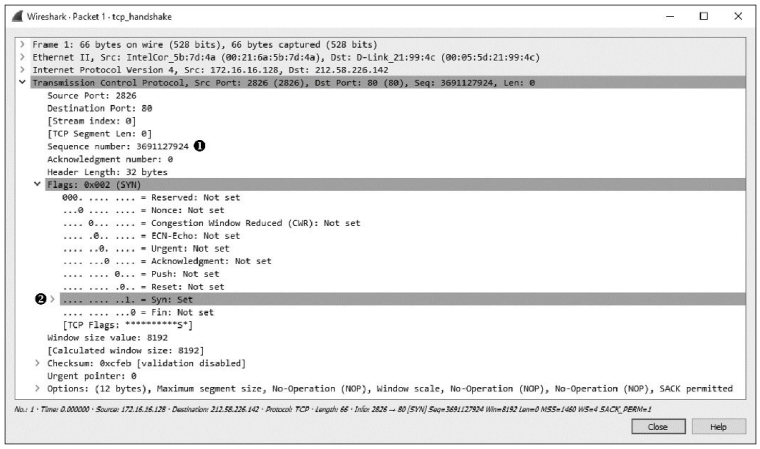

图 8-6　初始 SYN 数据包

握手中第二个数据包是从 212.58.226.142 发出的 SYN/ACK 响应（见图 8-7）。这个数据包也包含着这台主机的初始序列号（233779340）以及一个确认号（2691127925）。这个确认号比之前的那个数据包序列号大 1，因为这个域是用来表示主机所期望得到的下一个序列号的值的。

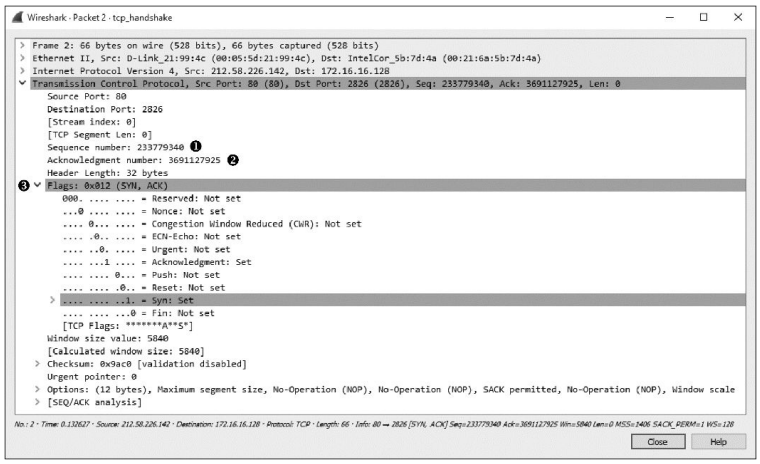

图 8-7　SYN/ACK 响应

最后的数据包是从 172.16.16.128（见图 8-8）发出的 ACK 数据包。这个数据包正如所期望的那样，包含着之前数据包确认号域所定义的序列号 3691127925。

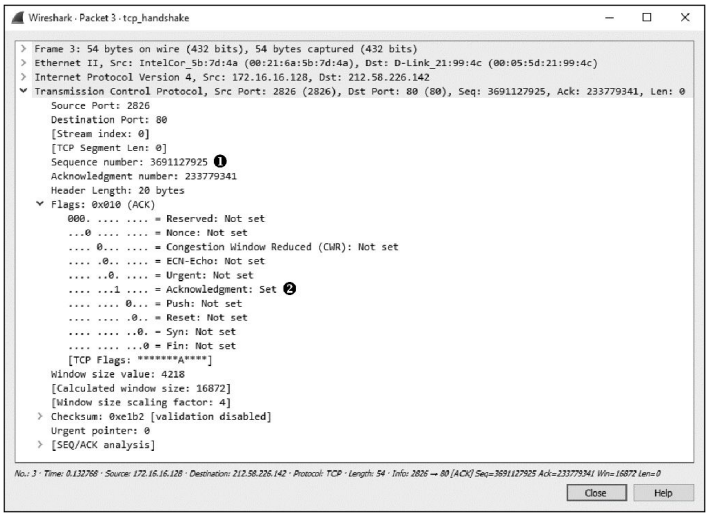

图 8-8　最后的 ACK 包

握手发生在每个 TCP 的通信序列之前。当在一个繁忙的捕获文件中搜索通信序列的开头时，序列 SYN、SYN/ACK、ACK 是一个很好的标志。