为提升性能而引入的的缓冲机制增加了网络应用程序的复杂性。例如采用fgets读取终端输入，当终端有多行数据输入时，select或epoll\_wait通知应用进程去读，**fgets会将多行数据读入应用缓冲区，但只返回第一行**，之后继续调用select或epoll\_wait监听终端输入，很显然，这里用户输入了多行数据，但是应用进程只处理了一行就继续等待终端输入。究其原因在于**select或epoll\_wait是不感知应用缓冲机制的**。