

Checksum Offload

Checksum Offload

A large part of the processing costs related to [TCP](#) is the generation and verification of the TCP checksum. Many Gigabit Ethernet chipsets include "on-board" hardware that can verify and/or generate these checksums. This significantly reduces the amount of work that has to be done by the system kernel on a CPU, especially when combined with other adapter/driver enhancements such as [Large-Send Offload](#). Checksum Offload is also part of [TCP Offload Engines \(TOEs\)](#), which move the entire TCP processing from the CPU(s) to the adapter. Checksum Offload requires special driver support and a kernel infrastructure that supports such drivers.

TCP Checksum Offload is the most common form of checksum offload, but of course it is possible to offload other checksums such as the UDP or SCTP checksums.

Some people (at HP?) abbreviate Checksum Offload as "CKO".

A possible issue with offloading checksums to the controller is that the integrity protection is less "end-to-end": If there are errors in the internal (bus) transmission of data from the host processor/memory to the adapter, the adapter will happily compute checksums on the corrupted data, which means that the corruption will go undetected at the receiver.

– Main.SimonLeinen -18 May 2008