SCTP Data Transmission

SCTP implementations must have flow and congestion control mechanisms according to RFC2960 , which ensures that SCTP can be introduced without problems in networks where TCP is in widespread use (see also Performance Evaluation of the Stream Control Transmission Protocol).

General Concepts

SCTP distinguishes different *streams* of messages within one SCTP association. This enables a delivery scheme where only the sequence of messages needs to be maintained per stream (partial in-sequence delivery) which reduces unnecessary head-of-line blocking between independent streams of messages (see also SCTP Streams).

SCTP operates on two levels:

- Within an association the reliable transfer of datagrams is assured by using a checksum, a sequence number and a selective retransmission mechanism. Without taking the initial sequence into account, every correctly received data chunk is delivered to a second, independent level.
- The second level realises a flexible delivery mechanism which is based on the notion of several independent *streams* of datagrams within an association. Chunks belonging to one or several streams may be bundled and transmitted in one SCTP packet provided they are not longer than the current path MTU.

Detection of loss and duplication of data chunks is enabled by numbering all data chunks in the sender with the so-called Transport Sequence Number (TSN). The acknowledgements sent from the receiver to the sender are based on these sequence numbers.
Retransmissions are timer-controlled. The timer duration is derived from continous measurements of the round trip delay. Whenever such a

retransmission timer expires, (and congestion control allows transmissions) all non-acknowledged data chunks are retransmitted and the timer is started again doubling its initial duration (like in TCP).
When the receiver detects one or more gaps in the sequence of data chunks, each received SCTP packet is acknowleged by sending a Selective Acknowledgement (SACK) which reports all gaps. The SACK is contained in a specific control chunk. Whenever the sender receives four consecutive SACKs reporting the same data chunk missing, this data chunk is immediately retransmitted (fast retransmit). Most up-to-date operating systems already support a similar optional extension to TCP (see RFC 2018).

Flow Control

SCTP uses an end-to-end window based flow and congestion control mechanism similar to the one that is well known from TCP (see RFC 2581 - TCP Congestion Control). The receiver of data may control the rate at which the sender is sending by specifying an octet-based window size (the so-called Receiver Window), and returning this value along with all SACK chunks.
The sender itself keeps a variable known as Congestion Window (short: CWND) that controls the maximum number of outstanding bytes (i.e. bytes that may be sent before they are acknowledged). Each receveived data chunk must be acknlowlegded, and the receiver may wait a certain time (usually 200 ms) before that is done. Should there be a larger number of SCTP packets with data received within this period of, every second SCTP packet containing data is to be acknowlegded at once by sending a SACK chunk back to the sender.

Selective Acknowledgement

The acknowledgements carry all TSN numbers that have been received by one side with them. That is, there is a so called **Cumulative TSN Ack** value, that inidicates all the data that has successfully been reassembled at the receivers side, and has either already been delivered to the receiving Upper Layer Process, or may readily be delivered upon request. Moreover, there are so-called **Gap Blocks** that indicate that segments of data chunks have arrived, with some data chunks missing in between. Should some data chunks have been lost in the course of transmission, they will either be retransmitted after the transmission timer has expired, or after four SACK chunks have reported gaps with the same data chunk missing. In the latter case, the missing data is retransmitted via the **Fast Retransmit** mechanism.
In case a retransmission occurrs which signals packet loss, the implementation must appropriately update congestion and flow control parameters.

Flow Control for Multihomed Endpoints

By default, all transmission is done to a previously selected address from the set of destination addresses, which is called the **Primary Address**. Retransmissions should be done on different paths, so that if one path is overloaded, retransmissions do not affect this path (unless the network topology is such that retransmissions hit the same point in the network where the data was dropped due to congestion). For certain network topologies that may have beneficial effects on overall throughput. Acknowledgements shall be sent to the transport address from which originated the data.
Should the active path have a high number of failures and its error counter exceed a boundary, the SCTP implementation notifies its upper layer process that the path has become inactive. Then a new primary path may (and probably should) be chosen by the application (for more information on this, see SCTP Multihoming).

Congestion Control

The congestion control behaviour of an SCTP implementation according to RFC2960 may have an impact where timely delivery of messages is required (i.e. transport of signalling data). However, this ensures the proper behaviour of SCTP when it is introduced on a large scale into existing packet switched networks such as the Internet. The congestion control mechanismns for SCTP have been derived from RFC 2581 - TCP Congestion Control), and been adapted for multihoming. For each destination address (i.e. each possible path) a discrete set of flow and congestion control parameters is kept, such that from the point of view of the network, an SCTP association with a number of paths may behave similarly as the same number of TCP connections.

Slow Start and Congestion Avoidance

As in TCP, SCTP has two modes, Slow Start and Congestion Avoidance. The mode is determined by a set of congestion control variables, and as already mentioned, these are path specific. So, while the transmission to the primary path may be in the Congestion Avoidance mode, the implementation may still use Slow Start for the backup path(s).
For successfully delivered and acknowledged data the congestion window variable (CWND) is steadily increased, and once it exceeds a certain boundary (called Slow Start Threshold, SSTRESH), the mode changes from Slow Start to Congestion Avoidance. Generally, in Slow Start, the CWND is increased faster (roughly one MTU per received SACK chunk), and in Congestion Avoidance mode, it is only increased by roughly one MTU per Round Trip Time (RTT).
Events that trigger retransmission (timeouts or fast retransmission) cause

the SSTHRESH to be cut down drastically, and reset the CWND (where a timeout causes a new Slow Start with CWND=MTU, and a Fast Retransmit sets CWND=SSTHRESH).

Path MTU Discovery

Since the Path MTU is such an important variable (influencing the congestion control), an SCTP implemetations should keep a variable for the estimate of the current Maximum Transmission Unit (Path MTU) for each path. How this is done, is closer described in RFC 1191 - Path MTU Discovery and RFC 1981 - Path MTU Discovery for IP Version 6.