

PacketReordering

Packet Reordering

The Internet Protocol (IP) does not guarantee that packets are delivered in the order in which they were sent. This was a deliberate design choice that distinguishes IP from protocols such as, for instance, ATM and IEEE 802.3 Ethernet.

Decomposition

Reasons why a network may reorder packets: Usually because of some kind of parallelism, either because of a choice of alternative routes ([Equal Cost Multipath, ECMP](#)), or because of internal parallelism inside switching elements such as routers. One particular kind of packet reordering concerns packets of different sizes. Because a larger packet takes longer to transfer over a serial link (or a limited-width backplane inside a router), larger packets may be "overtaken" by smaller packets that were sent subsequently. This is usually not a concern for high-speed bulk transfers - where the segments tend to be equal-sized (hopefully [Path MTU](#)-sized), but may pose problems for naive implementations of multi-media (Audio/Video) transport.

Impact on end-to-end performance

In principle, applications that use a transport protocol such as [TCP](#) or [SCTP](#) don't have to worry about packet reordering, because the transport protocol is responsible for reassembling the byte stream (message stream(s) in the case of [SCTP](#)) into the original ordering. However, reordering can have a severe performance impact on some implementations of [TCP](#). Recent [TCP](#) implementations, in particular those that support [Selective Acknowledgements \(SACK\)](#), can exhibit robust performance even in the face of reordering in the network.

Real-time media applications such as audio/video conferencing tools often experience problems when run over networks that reorder packets. This is somewhat remarkable in that all of these applications have [jitter buffers](#) to eliminate the effects of [delay variation](#) on the real-time media streams. Obviously, the code that manages these jitter buffers is often not written in a way to accommodate reordered packets sensibly, although this could be done with moderate effort.

Measurement

Packet reordering is measured by sending a numbered sequence of packets, and comparing the received sequence number sequence with the original. There are many possible ways of quantifying reordering, some of which are described in the IPPM Working Group's documents (see the [references](#) below).

The measurements here can be done differently, depending on the measurement purpose:

- a) measuring reordering for particular application can be done by capturing the application traffic (e.g. using the [Wireshark/Ethereal](#) tool), injecting the same traffic pattern via traffic generator and calculating the reordering.
- b) measuring maximal reordering introduced by the network can be done by injecting relatively small amount of traffic, shaped as a short bursts of long packets immediately followed by short burst of short packets, with line rate. After capture and calculation on the other end of the path, the results will reflect the worst possible packet reordering situation which may occur on particular path.

For more information please refer to the measurement page, available at [packet reordering - measurement site](#). Please notice, that although the tool is based on older versions of reordering drafts, the metrics used are compatible with the definitions from new ones.

In particular Reorder Density and Reorder Buffer-Occupancy Density can be obtained from [tcpdump](#) by subjecting the dump files to [RATtool](#), available at [Reordering Analysis Tool Website](#)

Improving reordering

The probability of reordering can be reduced by avoiding parallelism in the network, or by using network nodes that take care to use parallel paths in such a way that packets belonging to the same end-to-end flow are kept on a single path. This can be ensured by using a hash on the destination address, or the (source address, destination address) pair to select from the set of available paths. For certain types of parallelism this is hard to achieve, for example when a simple striping mechanism is used to distribute packets from a very high-speed interface to multiple forwarding paths.

References

- IETF IPPM Working Group: [official charter page](#) - [tools.ietf.org WG status page](#)
- [RFC 4737](#), *Packet Reordering Metric for IPPM*, A. Morton, L. Ciavattone, G. Ramachandran, S. Shalunov, J. Perser, November 2006
- [RFC 5236](#), *Improved Packet Reordering Metrics*, A. Jayasumana, N. Piratla, T. Banka, A. Bare, R. Whitner, June 2008

-- Main.SimonLeinen - 2004-10-28 - 2014-12-29

– Main.MichalPrzybylski - 2005-07-19

– Main.BartoszBelter - 2006-03-28