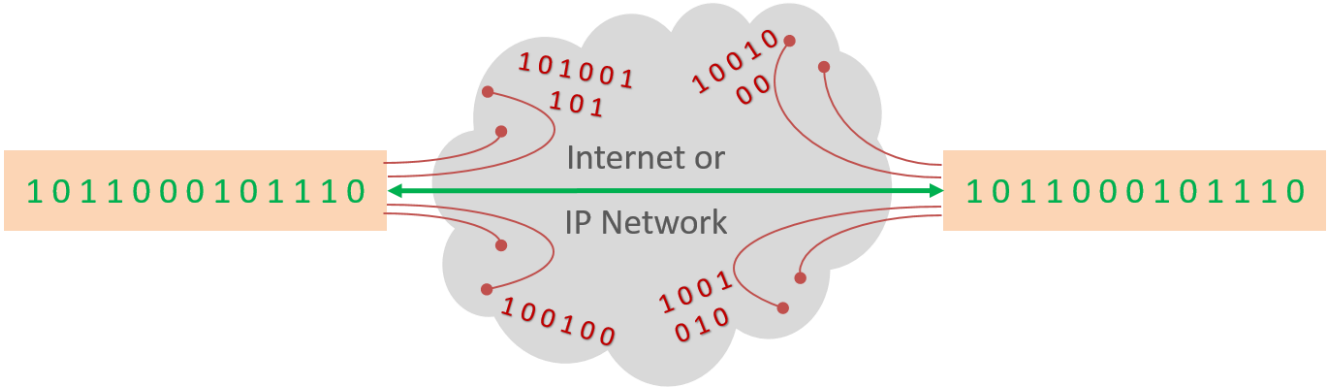
What Causes Packet Loss
IP Networks

(Rev 1.1)



Computer Modules, Inc.
DVEO Division
11409 West Bernardo Court
San Diego, CA 92127, USA
Telephone: +1 858 613 1818
Fax: +1 858 613 1815
www.dveo.com

Introduction

This Document

To stream high quality video, i.e. to transmit real-time video streams, over IP networks is a demanding endeavor and, depending on network conditions, may result in packets being dropped or "lost" for a variety of reasons, thus negatively impacting the quality of user experience (QoE). This document looks at typical reasons and causes for what is generally referred to as "packet loss" across various types of IP networks, whether of the managed and conditioned type with a defined Quality of Service (QoS), or the "unmanaged" kind such as the vast variety of individual and interconnected networks across the globe that collectively constitute the public Internet ("the Internet").

The purpose of this document is to encourage operators and enterprises that wish to overcome streaming video quality problems to explore potential solutions. Proven technology exists that enables transmission of real-time video error-free over all types of IP networks, and to perform live broadcasting of studio quality content over the "unmanaged" Internet!

Core Protocols of the Internet Protocol Suite

The Internet Protocol (IP) is the foundation on which the Internet was built and, by extension, the World Wide Web, by enabling global *internetworking*. The protocol embodies the ability to deliver or route datagrams ("packets") from source to destination across multiple IP network boundaries, traversing regions, countries and entire continents, based on a relatively simple addressing concept (the "IP address").

IP is by design a "connectionless" datagram protocol, and it is complemented by the connection-oriented Transmission Control Protocol (TCP), together referred to as TCP/IP. IP and TCP are the core protocols of the Internet protocol suite. Together they became the communications protocols at the heart of the global Internet as we know it, a development that the co-inventors of TCP/IP could hardly have imagined back in 1974 (Vint Cerf and Bob Kahn; both went on to become much honored internet pioneers).

In the Internet protocol suite, IP occupies a position in the Internet layer, above the bottom Link layer, whereas TCP is part of the Transport Layer, which sits above the Internet layer and below the Application layer. TCP/IP enables reliable delivery of a packet and its "encapsulated data" from sender to receiver, regardless of physical distance and however long it may take given prevailing network conditions.

To enable this, an IP packet consists of two components: the address section ("packet header"), and the data to be transported and ultimately delivered ("payload"). The address section holds a packet's source IP address, destination IP address, and metadata required for routing and delivery.

The role of IP is to deliver a packet from the sender ("source host") to the receiver ("destination host") based on the IP address in the packet header.

Network Reliability Assumptions and Issues

The design of the Internet protocols is based on the assumption that the network infrastructure is inherently unreliable at any single node or transmission link. However, it also assumes that the infrastructure is dynamic in terms of availability of network paths (links) and nodes. Furthermore, there is no overall network management authority to monitor or maintain links and nodes. The intelligence in the network is located primarily at the end points, that is at the data transmission origination and reception points, in order to minimize internal network complexity. Nodes in the transmission path simply route packets to the next known and reachable node matching information in the packet header for the destination host.

IP and TCP Characteristics

An inherent characteristic of a connectionless protocol such as IP is that it can only offer "best effort" delivery, and hence its service must be deemed unreliable in contrast to a connection-oriented protocol such as TCP. Network error conditions may result in, for example, corrupted packet payloads, outright packet loss, packet duplication and out-of-sequence delivery.

IP routing is dynamic, that is, each packet is treated independently. The network maintains no record ("state") based on the path taken by prior packets. Therefore, a group of packets, transmitted in a certain sequence, may reach their common destination via different paths that may exhibit different network link timing characteristics ("latency"), resulting in packets arriving out-of-sequence at the destination compared to their original order of transmission.

TCP, in contrast to IP, provides for a reliable, sequence-ordered, and error-checked delivery of packets between source and destination hosts communicating over an IP network.

TCP/IP vs. UDP: Protocol Comparison

Two Vastly Different Protocols

As mentioned earlier, TCP (TCP/IP in reality) emphasizes reliability over latency, which means that it will eventually deliver all packets regardless of network conditions and the time it may take to accomplish it. Major Internet applications such as the World Wide Web, email, Secure Shell (SSH) and File Transfer Protocol (FTP) rely on TCP/IP. Common to those applications is that reliability is more important than timeliness.

On the other hand, there are several types of time critical applications where low-latency delivery is of higher importance, and which simply cannot afford the protocol overhead of TCP/IP that may cause variable and unacceptable delays in packet delivery. Such applications may benefit from the lightweight User Datagram Protocol (UDP), a connectionless datagram service with minimum protocol overhead that puts reduced latency ahead of reliability. UDP lends itself very well for time critical applications such as streaming video, Voice over IP (VoIP), and online multiplayer games. However, UDP works best in uncongested networks or where real-time UDP traffic is given priority over data-centric TCP/IP ditto.

TCP/IP	UDP
Type: Connection-oriented protocol; it knows who the receivers are.	**Type:** Connection-less protocol; it does not care who the receivers are.
Reliability: Guaranteed delivery of every packet; will retransmit if ACK not received within a defined time-out period.	**Reliability:** No guarantee of delivery since there is no ACK mechanism; hence no retransmit if a packet is lost.
Ordering: Packets guaranteed to arrive in correct order.	**Ordering:** May arrive out of sequence, or duplicated; no automatic reordering.
Overhead: Heavyweight protocol. Positive ACKs required for every packet (block) adds overhead, together with resend requests for lost and wrong-order packets.	**Overhead:** Lightweight "fire and forget" design without requiring positive ACKs, and no connection tracking, keeps protocol nimble; ideal for real-time tasks.
Streaming: Multiple packets per read call.	**Datagrams:** One packet per read call.
Routing Protocol: Unicast only.	**Routing Protocols:** Unicast and multicast.
Uses: World Wide Web, email, File Transfer Protocol (FTP), Secure Shell (SSH).	**Uses:** Streaming (real-time) media, live video, VoIP, online multiplayer games.

Packet Loss Effects

Packet loss occurs when one or more packets transmitted over an IP network fail to arrive at their destination. Packet loss is typically caused by what is generally referred to as "network congestion," which in itself can have a number of actual causes. Packet loss is measured as the percentage of packets lost compared to packets transmitted.

In time critical applications that use UDP, such as streaming (real-time) video delivery, as well as VoIP and online multiplayer game applications, packet loss can affect the user experience even when the percentage of packets lost is just a fraction. For video, which is the most time sensitive and hence challenging application, visual artifacts result in high quality video even when packet loss is less than 1 percent of packets transmitted.

As mentioned, TCP/IP by design detects packet loss and performs retransmissions to ensure reliable messaging. TCP/IP will also reduce the throughput of a connection upon detection of packet loss, in order to avoid congestion and let the network "catch up." While this behavior may not adversely affect typical data-centric applications such as email, web browsing and file transfers, it can have a very undesirable impact on the user experience for real-time services, notably streaming video and online gaming, and therefore TCP/IP is not a suitable protocol for time critical uses. Unfortunately, TCP/IP's behavior can also cause reduced throughput for real-time protocols like UDP when sharing routers and buffers.

In more concrete terms, video playback may experience gaps due to delayed or lost packets unless there is a low-latency automatic packet recovery mechanism that can manage rapid resending of lost packets, and reorder packets that arrived out of sequence, so that no visible gaps will appear to the viewer. UDP, while ideal for real-time applications in uncongested or video optimized networks, does not have any built-in packet recovery functionality. It would be the responsibility of the application to manage that, and there are technologies available that can do just that. But first we will look further into possible causes of packet loss.

Possible Causes of Packet Loss

Unintentional Packet Loss

Network congestion is a generalized term for issues that cause a disorderly treatment or routing of packets (including no routing at all, i.e. packet loss). Simply put, if more packets are put through a network node than it can reasonably handle, i.e. when packets arrive for a continuous period of time at a rate higher than the node is able to route through, then packets will be dropped (i.e. not routed to their destinations). If a single node or link is limiting the throughput, it is known as a "bottleneck." The resulting node or link congestion may manifest itself in several ways:

- Packet Loss: The packet does not arrive at all, e.g. a router experiencing momentary overload
- Packets out of Sequence: Arriving in the wrong order, e.g. due to taking different paths with different latency
- Packet Delay Variation (PDV): Variations in arrival time

  The term PDV, a formal ITU term, is more commonly referred to as "jitter." It is defined as variations in network latency, i.e. variable end-to-end delay instead of constant latency. Jitter leads to variations in the delay between received packets, and it is often a result of multi-network "hops" that introduce variable latency.

A more detailed look will reveal several related causes that result in packets not being delivered as originally intended:

- Data must traverse a multitude of links with various types of network devices, with varying degrees of reliability
- Routers may experience momentary overloads, which results in dropped packets
- TCP packets may capture resources due to excess buffering, slowing the throughput after buffers fill up, which leads to UDP packets being dropped or cause unacceptable levels of latency and jitter for real-time applications (see "buffer bloat" below)
- Routers may not be configured for video in general and high-bitrate video in particular
- If wireless networks are used, signals may be marginally too weak or suffer frequency interference from other networks or equipment that transmit at the same frequency (compare 2.4 GHz home networks with cordless phones on the same frequency)
- Routers may have bugs and faulty software, perhaps an upgrade gone wrong
- Network hardware may be malfunctioning due to cabling issues (including rodents gnawing on cables!)
- Hardware may also experience a power supply failure, where cheap hardware is not equipped with dual Power Supply Units (PSU), or even an equipment premises power outage without backup generator availability

The list above mostly assumes *unintentional* packet loss as a result of accidental or other uncontrolled circumstances.

Intentional Packet Loss

Packet loss is not necessarily an indication of poor connection reliability or a bottleneck. Packet loss can also be *intentional* and used as a network management means to balance or prioritize available bandwidth between several competing senders as a particular router or link nears its maximum throughput capacity. In a situation where QoS is *rate limiting* a connection, packets may be intentionally dropped in order to slow down specific services to ensure available bandwidth for other services marked with higher importance (possibly as defined in a Service Level Agreement or SLA).

Potential Effects of TCP/IP and "Buffer Bloat" on Real-time UDP Traffic

The TCP *congestion avoidance algorithm* relies on packet drops to assess available bandwidth. It speeds up the data transfer until packets start to drop, then slows down the transmission rate. Packets are queued within a router buffer before being transmitted but if the buffer fills up, additional incoming packets will be dropped.

When a network node becomes congested, there are various queuing methods used to determine which packets to drop. Most basic networking equipment will use the classical FIFO (First In First Out) queuing mechanism for packets waiting to get routed through the node, and it will drop a packet if the queue is full at the time a new packet arrives ("tail drop"). This is exactly where slow moving TCP/IP data transmissions can cause serious problems for time critical UDP traffic due to a phenomenon aptly called "buffer bloat," which occurs when a congested network link will cause UDP packets to become queued in buffers shared with TCP/IP for too long. In a FIFO queuing system, overly large router buffers result in longer queues, and excess buffering of packets causes high latency and PDV (jitter), which in turn will reduce the overall network throughput.

The buffer bloat problem often stems from network devices configured with large buffers, which can cause otherwise very high-speed networks to reduce the throughput of real-time and interactive applications such as streaming video and online gaming. In such a situation, as the buffer is full of the packets of a TCP/IP stream, incoming UDP packets are then dropped or delayed, which is undesirable for any connection that needs real-time throughput, not least streaming video.

In older routers, buffers filled up quickly due to their much smaller size, which led to a faster state of packets loss after link saturation was reached. This allowed the TCP protocol to adjust quicker, and the buffer bloat problem would be insignificant. However, newer routers have buffers so large that they can hold megabytes of data, which means that several seconds may be required to empty such a buffer when congested. This causes TCP (and its congestion avoidance algorithm) that shares bandwidth with other protocols on a link to react very slowly and packets that share a buffer will stay there longer than acceptable for real-time applications.

This problem especially affects datagram protocols such as UDP, because all packets passing through a common buffer (with a single queue) will suffer the same delay or latency. While a higher latency is typically not a problem for TCP/IP applications, it will negatively impact UDP applications such as real-time video by causing visible artifacts.

Ironically, available total bandwidth may end up partly unused since some destinations configured to handle very high bitrates, like live HD video, may not be reached in a timely manner due to large buffers clogged with TCP data awaiting delivery to “slow data destinations”, thus causing UDP packet loss or unacceptable delays even though the total bandwidth could have accommodated all traffic if routers were better chosen or configured, or if the overall network was divided into data and video segments. Again, this is something that a video operator relying on third-party networks has little control over and will need to resort to other tools to overcome such challenges. Fortunately, there are technologies available today able to deal with even severe UDP packet loss – described later.

Packet Loss Remediation

The first step would be to understand why there is packet loss and where in occurs in an end-to-end network. Packet loss is obviously detected by application protocols such as TCP, but a human is often required to detect and, like a physician, diagnose the reason for packet loss. Routers may offer status pages or logs, where it is possible to determine the number or percentage of packets dropped during a specific time period, but a professional network admin usually prefers a purpose-built tool for remote detection and diagnosis.

The Internet Control Message Protocol (ICMP) is another important protocol of the Internet protocol suite. It is used by network devices to signal, for example, that a requested service is unavailable or that a host or router could not be contacted. ICMP can also send query messages through an "echo" functionality, where a special packet is transmitted that should produce a reply after a specified number of network hops, from whichever node received it. A tool such as MTR (*My traceroute*, a combination of two other tools, *ping* and *traceroute*) uses ICMP to provide a visual representation of the path packets are taking, and to measure response times and packet loss at each hop.

That said, some of the more obvious ways to avoid or overcome packet loss issues would include:

- Ensuring that network hardware and software is functioning per specifications
- Use a network monitoring system that can detect hardware malfunction and router overload
- Install higher capacity routers and network switches
- Configure routers with smaller buffers to avoid buffer bloat
- Increase bandwidth by combining network devices into load balanced clusters
- Transmit packets across multiple network paths and use different ISPs to increase likelihood that all packets will arrive (multipath transmission)
- In mixed TCP/IP and UDP networks, prioritize real-time traffic using UDP
- Use protocols with built-in packet recovery like TCP/IP, which guarantees that every packet will arrive however long it takes – however, this is not feasible for real-time applications like video and voice
- Add low-latency automatic packet recovery to UDP such as ARQ (Automatic Repeat Request) technologies to ensure “safe passage” of streaming video and other real-time applications

It should be obvious that many of above remedies are not practical for a video operator that relies on third-party networks and multi-hop segments, since it can’t control how individual network devices are configured and so on. That is why the final remedy is to apply a low-latency automatic packet recovery technology for UDP such as ARQ (see below).

The Relentless Growth of IP Video

Cisco Report Underscores IP Video Growth

With the transition of video transmission from RF-based to IP networks, including the use of IP-based protocols when transmitting over RF infrastructure, video applications are now increasingly IP-based, whether in studios, video processing head-ends, content distribution networks and home networks.

Video is also growing strongly as a percentage of all internet traffic and is expected to continue to do so for years to come, both as a result of the uncapped growth is user generated video uploads, and the desire of consumers to watch video anytime and anywhere, and in higher quality (increases resolution and bitrates, only partly mitigated by higher efficiency video coding techniques such as HEVC). According to a Cisco report, Cisco Visual Networking Index: Forecast and Methodology, 2015–2020: "Globally, IP video traffic will be 82 percent of all consumer Internet traffic by 2020, up from 70 percent in 2015. Global IP video traffic will grow threefold from 2015 to 2020, a CAGR of 26 percent. Internet video traffic will grow fourfold from 2015 to 2020, a CAGR of 31 percent."

That also means that real-time video is the focus for QA related efforts to ensure that consumers can enjoy the same kind of television quality experience for IP-delivered services that they are accustomed to from traditional broadcasting over RF-based networks. Automatic and low-latency packet recovery and packet reordering is the center of attention and several companies have solutions to address that, with DVEO being a leader in the field.

Fixing Packet Loss – Introduction to ARQ Technology

Automatic **R**epeat re**Q**uest, sometimes also referred to as **A**utomatic **R**epeat **Q**uery, is a type of technology designed to achieve dynamic and low-latency error correction for the benefit of real-time applications, especially Live Video. ARQ is the perfect complement to UDP, a light weight and video centric protocol based on the "fire and forget" principle.

UDP in combination with ARQ is the enabler of reliable video delivery over the Internet. The pair achieves automatic packet recovery utilizing variable and adaptive processing, unlike TCP/IP's fixed overhead, thereby fulfilling the exacting demands of high-quality streaming video.

DVEO DOZER™ ARQ Technology

DOZER ARQ delivers real-time video error-free over unreliable network segments. The DVEO patented ARQ packet recovery algorithms will fix UDP packet loss, correct for jitter, de-duplicate and reorder packets. Defining characteristics:

- Robust: Reliable MPEG-2, H.264, H.265 and file delivery
- Flexible: Point-to-point or point-to-multipoint
- Secure: All transmissions are AES-128 encrypted
- Remote Management: SNMP, and secure web access (SSH)

Winner of Society of Broadcast Engineers (SBE) Technology Award 2014 for DOZER IP video traffic smoothing technology.

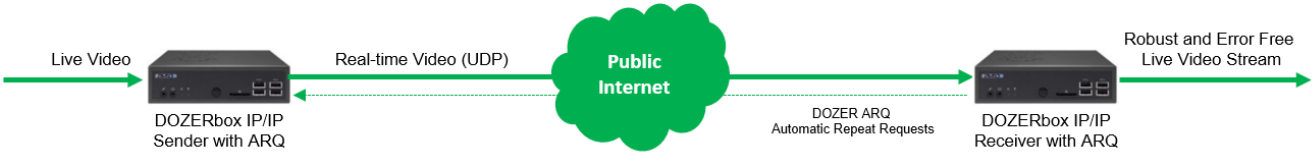


Figure 1: DOZER ARQ Pair Enables Error-free Live Video over the Internet

Who Benefits from DOZER ARQ?

- ✓ Broadcasters and Affiliates
- ✓ Live Event Producers and Electronic News Gathering
- ✓ Studios, Programmers, Content Providers and Aggregators
- ✓ Enterprises, Institutions and Government

Download the DOZER ARQ datashet here.

Anybody who wants to
Transmit Live, Studio-Quality Video over the Internet while Saving Big $$$
compared to traditional alternatives

Contact Us

Please contact DVEO for a no-obligation consultation on *how to fix packet loss*: +1 858 613-1818 or www.dveo.com,