UDP

UDP (rfc768) is an unreliable TL protocol designed to provide a service with minimum overhead. Its primary use is to transmit short messages, in particular messages that have a small time–to–live.

UDP does not use connections; every segment ("datagram") is independently sent, routed, and received. If a group of datagrams forms a sequence, it is up to the AL to order them appropriately.

UDP has no concept of feedback, hence there are no acknowledgments. The only guarantee offered by UDP is that every packet handed by it to the AL is correct (size and checksum).

UDP

A datagram has an 8 byte header followed by a variable–length payload. The header:

Source port: port of the sender process.

Destination port: port of the receiving process.

Length of the datagram (header and data).

Checksum: of the datagram header, imaginary IP header, and datagram data. Optional; if not used, set to 0.

Each field is 16 bits long.

Datagram format

Source port	Dest. port
Length	Checksum
Datagram body	

How does UDP work?

- How does it provide a stream connection?
- How does it inform the AL that it dropped a datagram?
- How does it react to a duplicate datagram?
- How does it react to a datagram arriving out of order?
- What does it do with datagrams that fail the checksum test?
- How does it handle application–layer messages that are longer than 64KB?

When to UDP?

- Periodic data collection from remote sensors (etc.).
- Periodic data dissemination, such as updates to routing tables (RIP, OSPF).
- Real–time traffic, including voice and video.
- Request–response, such as DNS.
- Online games.
- Multicast—not available in TCP.

Packet loss is bad?

An interesting report lists some of the myths about packet loss:

- Most loss is caused by transmission errors, gamma rays, etc.
- There is no loss in networks that are operating properly.
- Loss only happens in networks (i.e. not in hosts).
- All packet loss is bad.

Loss is mainly caused by buffer overflow. Packet loss is very beneficial in detecting congestion in TCP.

UDP security attacks

UDP is quite robust, mainly because of its simplicity (and the absence of connections to break into).

There are two popular attacks that use UDP (many other possibilities exist, but hackers do not take pride in breaking the breakable):

- UDP flooding causing congestion and DoS. This attack is trivial: simply flood the network with useless UDP datagrams, possibly all going to a particular destination.
- The diagnostic–port attack is a variation of flooding. The diagnostic ports exist (in routers in particular) for the so–called small servers that assist in troubleshooting (examples: echo in port 7, chargen in port 19 and discard in port 9). If a router is flooded with packets sent to one of these ports, it may become overwhelmed or even crash.

Real nasty attack

A nasty variation of the diagnostic–port attack on echo is to flood while spoofing the sender's address. This will cripple both the router and some innocent host (the spoofed address) which will also be overwhelmed with replies from the router.

Resource Reservation Protocol RSVP

The main Transport Layer protocols (TCP and UDP provide a transport service that is either reliable or reliable but offer no QoS provisions. This is obvious for UDP but an addition to TCP that would provide some QoS guarantees is welcome.

RFC2205 describes RSVP, a protocol that operates in conjunction with IP and potentially provides a way of reserving resources inside the network cloud; exclusive use of these resources provides a suitable quality of service.

The main weakness of RSVP lies in expecting IP entities (routers) to be able to perform resource management (mainly bandwidth but also buffer space, priority assignments, etc.).

Data flow

RSVP operates on session look–alikes called data flows. A data flow is defined as a sequence of packets (streams or datagrams) sharing the same:

- Transport Layer protocol.
- Source address.
- Destination IP address (port number is optional). This can be a multicast address.
- Quality of service as specified by the flow specification.

A data flow is unidirectional.

Quality of service

is specified by 3 parameters:

- Service class (3 possibilities).
- Reservation specification parameter which defines the resources needed to satisfy the desired quality of the flow.
- Traffic specification parameter describing the type of the data flow.

Service class

Three options:

Best–effort: provide reliable delivery with no packet loss without any restrictions on time delay.

Rate–sensitive: guarantee a minimum transmission rate (sometimes equated with constant bit rate). Often used in sessions involving audio transmissions (MPEG–1 AudioLayer–3 = MP3) or in cheap video (teleconferencing).

Delay–sensitive: guarantee an upper bound on delay for a variable bit rate flow. Typically used in transmitting quality video and audio (MPEG–2 expects a delivery rate of 25 or 29.97 frames per second, with the size of frames varying greatly).

Operation of RSVP

The basic two functions of RSVP are: path setup (not circuit) and path teardown.

A path is built by the exchange of 2 RSVP packets:

Path packet: sent by the source host to the destination address of the flow. It contains the sender's proposed flow specification, which all the routers on the way store internally before forwarding the packet. The path itself is obtained from its local routing protocols (protocols auxiliary to IP).

Reservation packet: is sent back by the each of the destination hosts to the sender—along the same path as the one travelled by the path packet. It contains the destination's flow specification.

As a reservation packet moves upstream, each router makes the appropriate resource reservations. If it is impossible to satisfy the flow specification, the router may either reject it (and send an error packet downstream) or modify the specification, reserve the (modified) set of resources and forward the reservation packet upstream.

If a reservation (changed or not) is made, the router sends a confirmation downstream.

The above behaviour applies to the routers that claim to understand RSVP. Those routers that do not understand RSVP will blindly forward the reservation and do nothing[a] (i.e. apply the regular IP best–effort service).

[a] This is known as tunnelling.

Teardown packet

can be sent by either end of the flow and results in the release of all the resources reserved for the flow.