

## Datagram Control Protocol (DCP)

Sally Floyd, Mark Handley, Eddie Kohler, Jitendra Padhye

<http://www.aciri.org/kohler/dcp/>

## Question

- ♦ How best to deploy congestion-controlled applications that require flow-like unicast semantics, but don't want absolute reliability?
  - Streaming Media
  - Internet Telephone
  - Unicast Multimedia Conferencing

## Issues

- ♦ Different apps want different congestion control dynamics (eg AIMD, TFRC).
- ♦ Concern about poor application-level congestion control implementations
- ♦ Firewalls and NATs don't co-exist very well with UDP.
  - Lack of explicit flow setup and teardown.
  - Lack of well-known ports because UDP setup is usually server-to-client.
- ♦ Concerns about deploying ECN with UDP applications.

## Datagram Control Protocol (DCP)

Key Features:

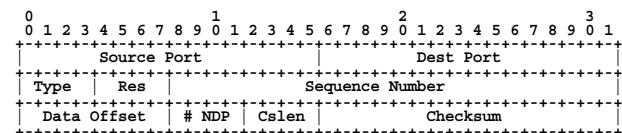
- ♦ Unreliable transport protocol, intended for kernel implementation on top of IP.
- ♦ Connection Setup:
  - Three-way handshake for connection setup.
  - "SYN" has service identifier in addition to src and dst ports.
  - Mandatory support for SYN cookies.
  - Allows negotiation of congestion control algorithm and parameters.
  - Sender-based algorithms can be added with support only at the sender.

## Datagram Control Protocol (DCP)

Key Features:

- ♦ Connection Progress:
  - Bidirectional asymmetric protocol; can use different options (such as CC) in each direction.
  - Provides packet sequence numbers and acks for congestion control.
  - In most uses, can also provide a reliable indication to the app as to which application packets arrived.
  - Supports ECN.
- ♦ Teardown handshake
  - Two or three way handshake on teardown (depending on initiator)
  - Time-wait state held on client.

## Common DCP Packet Header

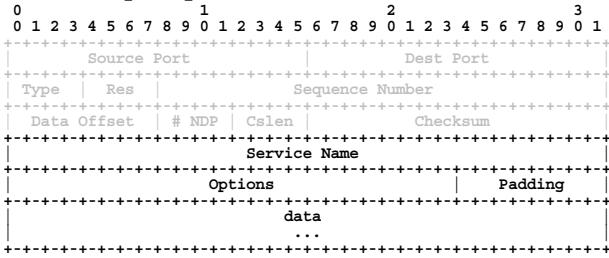


### Packet types:

- 0 DCP-Request packet.
- 1 DCP-Response packet.
- 2 DCP-Data packet.
- 3 DCP-Ack packet.
- 4 DCP-DataAck packet.
- 5 DCP-CloseReq packet.
- 6 DCP-Close packet.
- 7 DCP-Reset packet.

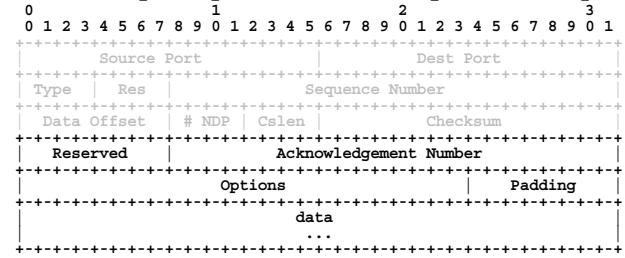
## DCP-Request Packet Format

- ♦ A DCP-Request packet is used to initiate a connection.



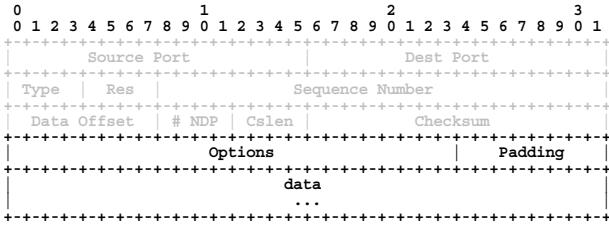
## DCP-Response

- ♦ A DCP-Response packet is sent in response to a request.



## DCP-Data, DCP-Ack, and DCP-DataAck

- ♦ The payload data in a DCP connection is sent in DCP-Data and DCP-DataAck packets. DCP-Data packets look like this:



- ♦ DCP-DataAck packets have an additional 32 bits including a 24 bit ACK field.

## Questions?

- ♦ Is a new transport protocol the best way to satisfy these requirements?
  - Can always use UDP.
  - SCTP is also in the process of gaining an unreliable mode.
- ♦ If so, did we get the basic feature set right?
- ♦ Interaction with RTP.
  - Both have a sequence space; we'd prefer not to have two in the same packet.
  - Use (or not) of SSRC.

## Feature Negotiation

- ♦ DCP provides a general purpose feature negotiation mechanism.
  - Multi-way handshake allows either side to initiate.
  - Eg. ask/answer, choose/ask/answer, ask/choose/ask/answer
- ♦ Primary use is negotiation of congestion control algorithms.
  - Separate feature negotiation happens for each direction.
- ♦ There are currently three DCP congestion control specifications:
  - TCP-Like AIMD congestion control
  - TFRC congestion control
  - Single-window congestion control
- ♦ Can also negotiate the use of an ACK vector to provide feedback of exactly which packets arrived.

## Drafts

- ♦ Datagram Control Protocol (DCP)
  - draft-kohler-dcp-00.txt,.ps
- ♦ Profile for DCP Congestion Control ID 0:Single-Window Congestion Control
  - draft-kohler-dcp-ccid0-00.txt,.ps
- ♦ Profile for DCP Congestion Control ID 2:TCP-like Congestion Control
  - draft-floyd-dcp-ccid2-00.txt,.ps
- ♦ Profile for DCP Congestion Control ID 3:TFRC Congestion Control
  - draft-padhye-dcp-ccid3-00.txt,.ps

