Signalling the UDP Encapsulation for DCCP

draft-ietf-dccp-udpencap-07.txt

Colin Perkins
Gorry Fairhurst
Background

- DCCP is a relatively new transport protocol
  - Connection oriented, client-server, unreliable, datagram transport
  - RFCs 4340 – 4342

- Mapping of RTP onto DCCP defined in RFC 5762
  - Signalling uses a new `proto` in SDP `m=` line to indicate DCCP, plus the SDP connection-oriented media extensions [RFC 4145]

```
v=0
o=alice 1129377363 1 IN IP4 192.0.2.47
s=-
c=IN IP4 192.0.2.47
t=0 0
m=video 5004 DCCP/RTP/AVP 99
a=rtcp-mux
a=rtpmap:99 h261/90000
a=dccp-service-code:SC=x52545056
a=setup:passive
a=connection:new
```

Example SDP offer
DCCP Encapsulation into UDP

• An encapsulation of DCCP in UDP is defined in draft-ietf-dccp-udpenccap-07

  • Wraps DCCP packets in UDP packets, to ease NAT traversal

```
+-----------------------------------+                     +-----------------------------------+
|     IP Header (IPv4 or IPv6)      |  Variable length |                                  |
+-----------------------------------+                     +-----------------------------------+
|            UDP Header             |  8 bytes         |                                  |
+-----------------------------------+                     +-----------------------------------+
|       DCCP Generic Header         |  12 or 16 bytes  |                                  |
+-----------------------------------+                     +-----------------------------------+
| Additional (type-specific) Fields | Variable length (could be 0) |
+-----------------------------------+                     +-----------------------------------+
|            DCCP Options            | Variable length (could be 0) |
+-----------------------------------+                     +-----------------------------------+
|      Application Data Area        | Variable length (could be 0) |
+-----------------------------------+                     +-----------------------------------+
```

• A simple shim-layer encapsulation
• In WG last call in the DCCP working group
Open Issue: Signalling

- How best to signal this in SDP?
  - For DCCP-in-UDP, need to signal that encapsulation is needed, and which UDP port is used
  - Some servers may support only native DCCP, some only DCCP-in-UDP, and some both

- Possible choices:
  - Define an SDP attribute to signal encapsulation
  - Define a new SDP proto for the m= line for DCCP-in-UDP
  - Other...?

- How to deal with NAT traversal?
Option 1: Attribute Indicates Encapsulation

- Signal DCCP as the protocol in the m= line
- Define a “a=dccp-in-udp:” attribute to indicate that UDP encapsulation is to be used, and to specify the port
  - This is what’s in the draft now

- Works well if DCCP servers generally support native DCCP, but some also support UDP encapsulation
- Inappropriate for DCCP servers that only support UDP encapsulation – which may be most of them

```plaintext
v=0
o=alice 1129377363 1 IN IP4 192.0.2.47
s=-
c=IN IP4 192.0.2.47
t=0 0
m=video 5004 DCCP/RTP/AVP 99
a=rtcp-mux
a=rtpmap:99 h261/90000
a=dccp-service-code:SC=x5254505
a=dccp-in-udp:9999
a=setup:passive
a=connection:new
```
Option 2: New $m=\text{proto}$ value

- Can define DCCP encapsulated in UDP as a new protocol
- Define an attribute to specify the UDP port on which the server is listening
- How to signal a server that supports both native and UDP-encapsulated DCCP?
- Is this the right approach, or should we do something else?

```
v=0
o=alice 1129377363 1 IN IP4 192.0.2.47
s=-
c=IN IP4 192.0.2.47
t=0 0
m=video 5004 DCCP-UDP/RTP/AVP 99
a=rtcp-mux
a=rtpmap:99 h261/90000
a=dccp-service-code:SC=x52545056
a=setup:passive
a=connection:new
a=dccp-udp-port:9999
```
Interactions with NAT Traversal, ICE, etc.

• Signalling the UDP port only works if there is no NAT between client and server

• How do any of these mechanisms interact with signalling for NAT traversal?
  • Do we need to integrate with ICE? If so, how?
Conclusion

- Other feedback welcomed – either on MMUSIC if relating to the signalling, or to the DCCP WG

- Hoping to wrap-up this work quickly