RTP and the Datagram Congestion Control Protocol

Colin Perkins

draft-perkins-dccp-rtp-02.txt

UNIVERSITY
of
GLASGOW

Talk Outline

- Review of changes since -01
  - RTCP and congestion control
  - Other changes
- Open issues
- Future direction

RTCP and Congestion Control

- RTP uses a configured nominal "session bandwidth" to determine the RTCP reporting interval
  - Scales per-participant reporting interval so that aggregate RTCP bandwidth is 5% of nominal session bandwidth
  - However, actual session bandwidth might be significantly different from the nominal configured bandwidth, when using rate adaptive codec over DCCP
    - Also actual bandwidth might be asymmetric, yet RTCP bandwidth assumed symmetric

- Two options:
  - Use nominal bandwidth, accepting that RTCP interval may be inaccurate
  - Modify RTCP rules to account for actual session bandwidth ⇒ complex

RTCP and Congestion Control

- Proposal:
  - Use nominal bandwidth, accepting that RTCP interval may be inaccurate
  - Delay RTCP packet transmission, if necessary to follow congestion control
    - Use actual send time as start of interval when calculating next send time

- Implications:
  - If DCCP channel capacity significantly below nominal session bandwidth:
    - RTCP packets may be delayed sufficiently to cause participants to time out
      - If packets delayed by more than five reporting intervals
    - But:
      - Nominal session bandwidth chosen based on codec capabilities; RTCP traffic small fraction (typically 5%) of the total
      - If nominal bandwidth much larger than the available bandwidth, session likely not usable due to constraints on media, before RTCP constraints problematic
      - Can renegotiate session parameters to use lower quality codec
  - If DCCP channel capacity significantly above nominal session bandwidth:
    - Safe, although RTCP underperforms

Other Changes in -02

- Clarified that the use of zero-length DCCP-Data replaces all types of application level keep-alive, not just RTP no-op packets
  - i.e. no need for in-call STUN keep-alive
- Partially fixed ABNF
  - Matches SDP ABNF case sensitivity fixes in RFC 4566
  - More fixes needed (to be covered later)
- Registered standard DCCP ports 5004 (“DCCP RTP”) and 5005 (“DCCP RTCP”)

Open Issues

- NAT keep-alive frequency
- Guidance on implementation of congestion control
- Interactions with different RTP profiles
- Security and partial checksums
- ABNF for DCCP service code attribute

NAT Keep Alive Frequency

- The draft states that zero-length DCCP-Data packets SHOULD be used as a keep-alive
- Doesn’t mandate how often the keep-alive packets are sent
  - Too often is wasteful of capacity
  - Not often enough allows NAT bindings to timeout

- ICE recommends a keep-alive be sent once every 15 seconds
  - draft-ietf-mmusic-ice-09.txt section 7.12

- Recommend RTP-over-DCCP follow this practice

Implementation of Congestion Control

- Section 4.1 lists “Provide more guidance on implementation of congestion control within an RTP application” as an open issue

- Would prefer not to do so in this draft:
  - Implementation of congestion control within RTP applications is still an evolving area
  - Not clear we can give useful guidance
  - Any guidance we give will likely need to be updated in light of experience

- Recommend:
  - Reference the DCCP user guide and TFRC media guide drafts to provide guidance on implementation of congestion control in RTP applications
  - Do not provide explicit guidance here

Interactions with RTP Profiles

- Section 4.5 states:

  > "In general, there is no conflict between new RTP Profiles and DCCP framing, and most RTP profiles can be negotiated for use over DCCP. The only potential for conflict occurs if an RTP profile changes the RTCP reporting interval or the RTP packet transmission rules, since this may conflict with DCCP congestion control."

- Gorry Fairhurst asked "The only potential conflict occurs if" – is this really so, or would it better to say the only "known"?
  - Neither – noticed other possible conflicts:
    - Between the RTP/SAVP profile and partial checksums (see next slide)
    - Between DCCP and profiles which mandate particular lower layer transports
  - Will explain these conflicts, and add wording to say that these are the only known conflicts, but that others may be possible

Security and Partial Checksums

- Basic RTP provides some minimal security
  - Encryption of RTP and RTCP packets using DES in CBC mode
- SRTP improves encryption, adds integrity protection
  - Uses AES in counter mode for encryption by default
  - Uses HMAC-SHA1 for integrity protection by default
  - Integrity protection optional, but strongly recommended

- Interactions between security and partial checksums:
  - Integrity protection and the use of partial checksums to deliver corrupt packets clearly conflict
  - Bit errors propagate with DES encryption ⇒ partial checksums useless; don’t propagate with counter mode AES
    - *Might* be able to use SRTP with partial checksums, if integrity protection disabled
  - Need to add some discussion of issues; get review by security area folks

ABNF for DCCP Service Code Attribute

- The ABNF for the “a=dccp-service-code:” attribute reads:

```
dccp-service-attr = %x61 "=dccp-service-code:" service-code
service-code      = hex-sc / decimal-sc / ascii-sc
hex-sc            = "SC=x" *HEXDIG
decimal-sc        = "SC="  *DIGIT
ascii-sc          = "SC:"  *sc-char
sc-char           = %d42-43, %d45-47, %d63-90, %d95, %d97-122
```

- However, SDP attribute names and values are case sensitive; ABNF is not

- Need to fix ABNF using by explicit character codes throughout
  - Trivial fix; common oversight in many SDP-related drafts

Future Directions

- Only minor open issues remain; expect updates to address these over the summer
- Accept as work item?