DOTS Q-Block interop testing report

IETF111 DOTS WG

Kaname Nishizuka/NTT Communications

Jon Shallow

Interop Plan

- Target Draft
  - draft-ietf-core-new-block-14
- 2 independent implementations
  - go-dots (https://github.com/nttdots/go-dots) by NTT
  - Jon’s Implementation
- 2 days of preliminary interop testing of Q-Block2
  - DOTS server to DOTS client only (Section 10.2)
- Quick recap
  - Both ends use libcoap library forked by Jon
  - Successfully communicated each other
  - even with certain rate of packet loss

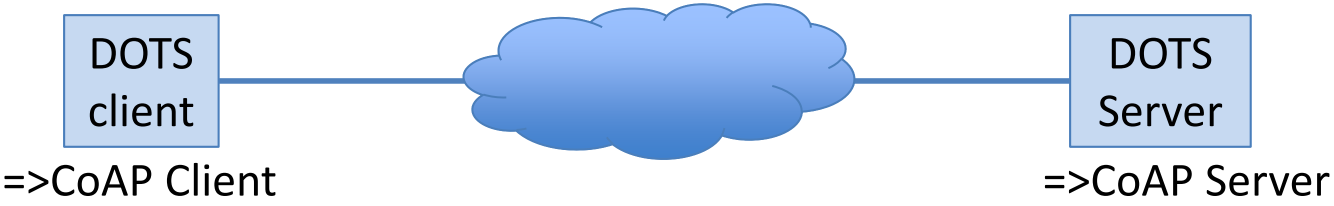
Design of Q-Block

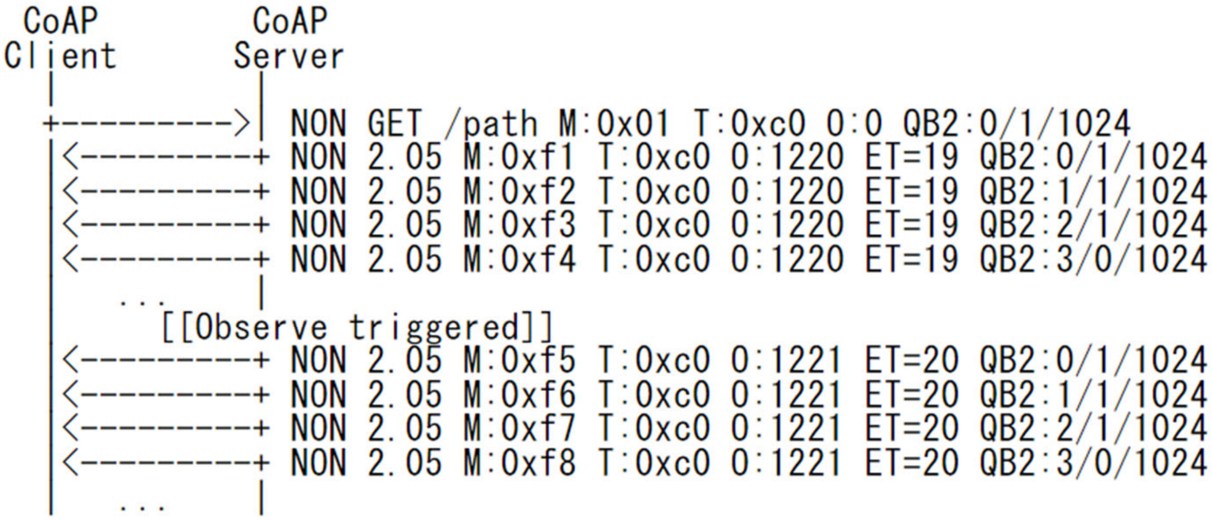
Purpose

- supporting Non-confirmable messages
  - Needed for (large) telemetry updates
  - NON because of DDoS uni-directional network pipe loss
- fewer packet interchanges
- faster recovery (should any of the blocks get lost in transmission)

Testing Environment

(without loss)



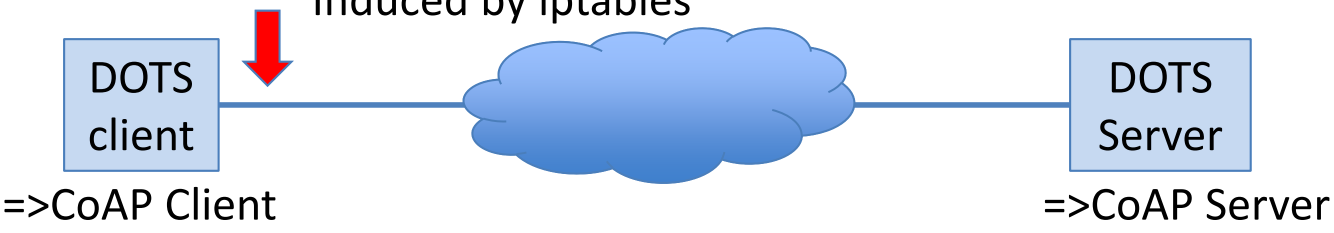


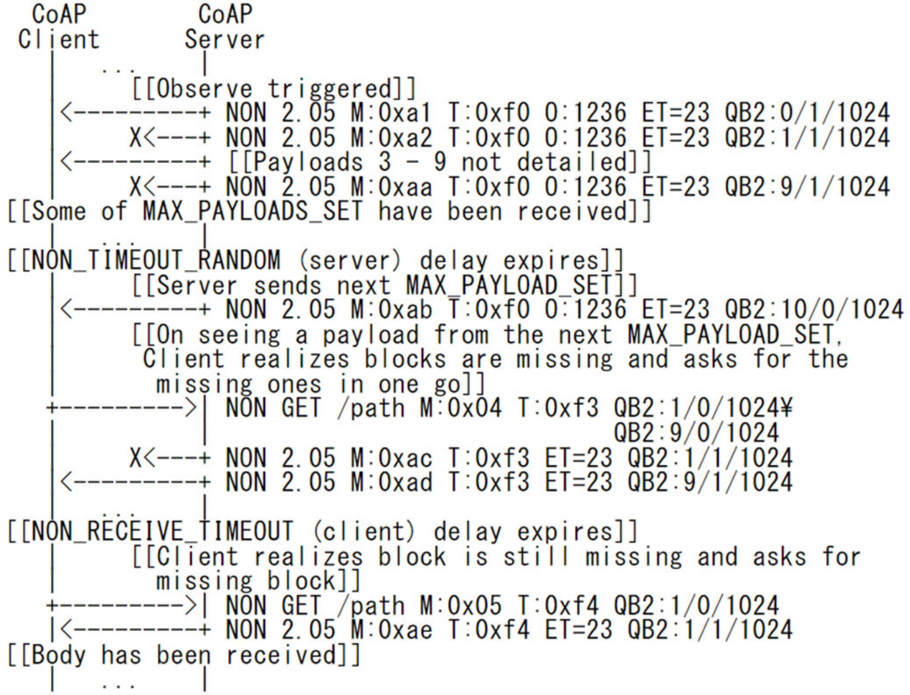
```
 CoAP        CoAP
Client      Server
  |           |
  +---------->| NON GET /path M:0x01 T:0xc0 O:0 QB2:0/1/1024
  |<----------+ NON 2.05 M:0xf1 T:0xc0 O:1220 ET=19 QB2:0/1/1024
  |<----------+ NON 2.05 M:0xf2 T:0xc0 O:1220 ET=19 QB2:1/1/1024
  |<----------+ NON 2.05 M:0xf3 T:0xc0 O:1220 ET=19 QB2:2/1/1024
  |<----------+ NON 2.05 M:0xf4 T:0xc0 O:1220 ET=19 QB2:3/0/1024
  |    ...    |
  |      [[Observe triggered]]
  |<----------+ NON 2.05 M:0xf5 T:0xc0 O:1221 ET=20 QB2:0/1/1024
  |<----------+ NON 2.05 M:0xf6 T:0xc0 O:1221 ET=20 QB2:1/1/1024
  |<----------+ NON 2.05 M:0xf7 T:0xc0 O:1221 ET=20 QB2:2/1/1024
  |<----------+ NON 2.05 M:0xf8 T:0xc0 O:1221 ET=20 QB2:3/0/1024
  |    ...    |
```

Figure 8: Example of NON Notifications with Q-Block2 Option (Without Loss)

Testing Environment

(with loss)





```
  CoAP        CoAP
 Client      Server
   |    ...    |
   |  [[Observe triggered]]
   |<---------+ NON 2.05 M:0xa1 T:0xf0 O:1236 ET=23 QB2:0/1/1024
   |     X<---+ NON 2.05 M:0xa2 T:0xf0 O:1236 ET=23 QB2:1/1/1024
   |<---------+ [[Payloads 3 - 9 not detailed]]
   |     X<---+ NON 2.05 M:0xaa T:0xf0 O:1236 ET=23 QB2:9/1/1024
[[Some of MAX_PAYLOADS_SET have been received]]
   |    ...    |
[[NON_TIMEOUT_RANDOM (server) delay expires]]
   |  [[Server sends next MAX_PAYLOAD_SET]]
   |<---------+ NON 2.05 M:0xab T:0xf0 O:1236 ET=23 QB2:10/0/1024
   |  [[On seeing a payload from the next MAX_PAYLOAD_SET,
   |    Client realizes blocks are missing and asks for the
   |     missing ones in one go]]
   +--------->| NON GET /path M:0x04 T:0xf3 QB2:1/0/1024¥
   |          |                             QB2:9/0/1024
   |     X<---+ NON 2.05 M:0xac T:0xf3 ET=23 QB2:1/1/1024
   |<---------+ NON 2.05 M:0xad T:0xf3 ET=23 QB2:9/1/1024
   |    ...    |
[[NON_RECEIVE_TIMEOUT (client) delay expires]]
   |  [[Client realizes block is still missing and asks for
   |     missing block]]
   +--------->| NON GET /path M:0x05 T:0xf4 QB2:1/0/1024
   |<---------+ NON 2.05 M:0xae T:0xf4 ET=23 QB2:1/1/1024
[[Body has been received]]
   |    ...    |
```

Figure 10: Example of NON Notifications with Q-Block2 Option (Blocks Recovery)

Result

Without loss

- Successfully received entire body larger than MAX_PAYLOADS payloads
  - Every MAX_PAYLOADS count gives Congestion Control pause if no ‘Continue’ response
- Fewer packets (compared with Block2 (which normally requires Confirmable))

With loss

- Successfully recovered entire body even with 1%, 3%, 5%, 10% packet loss rate
- Fewer packets: Reclaim of missing blocks in one go
- Found a few libcoap bugs (now fixed)

libcoap bugs and discussion (1/2)

Issue:

It was not clear before the interop which request method should be used for asking for missing blocks (e.g. PUT triggered a large blocked response)

Conclusion:

When requesting the additional blocks (Block2) or requesting the missing blocks (Q-Block2) then the request method (+ appropriate (Q-)Block2 options) is the same for the next blocks as the initial request - even if it was a PUT

libcoap bugs and discussion (2/2)

Issue:

Initial GET for the entire body (NUM is zero and Mbit is set) could be misunderstood by the CoAP server as a request for the remaining missing blocks of the previous blocked response.

Conclusion:

GET (NUM is zero and Mbit is set) should be treated as a request for the entire (new or refreshed) body.

[draft 4.4 Using the Q-Block2 Option
NUM is zero: This is a request for the entire body.]

Questions?

Thank You