

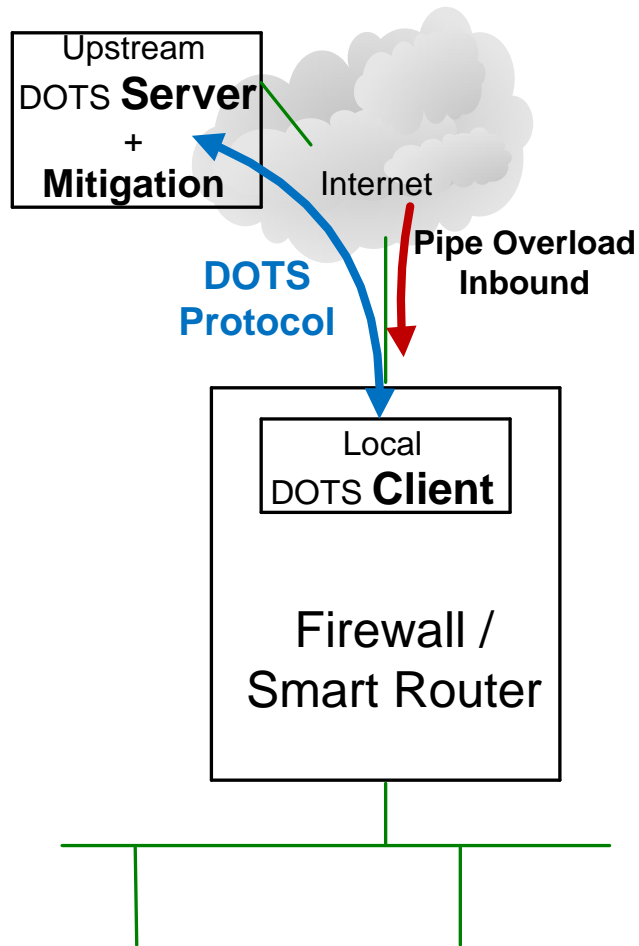
# New CoAP Block-Wise Transfer Options draft-bosh-core-new-block

CoRE virtual interim, 10<sup>th</sup> June 2020

**Mohamed Boucadair**

**Jon Shallow**

# Sample Target Deployment



- DDoS Open Threat Signalling (DOTS)
- DOTS: App – CBOR – CoAP – DTLS – IP
- Client requests mitigation (NON)
- Server updates with simple DOTS mitigation status (NON)
- Inbound Pipe Overload
  - Clients can still request mitigations
  - Mitigation should be able to control pipe overload
- See [RFC8782](https://www.rfcs.org/rfc/8782) for more details

# Summary of Updates Since -00

- Applicability Scope added
- Guards to prevent a CoAP agent from overloading the network
  - PROBING\_RATE clarification
  - MAX\_PAYLOADS defined, with a default value of 10

Congestion Control

- Detailed description of Block3/Block4 Option
  - Block3 no longer repeatable
  - New CoAP Response Code for missing blocks
- Block3/Block4 RFC8613 definitions included
- Streaming support (*general use*)
- Text tidy up

Protocol Machinery

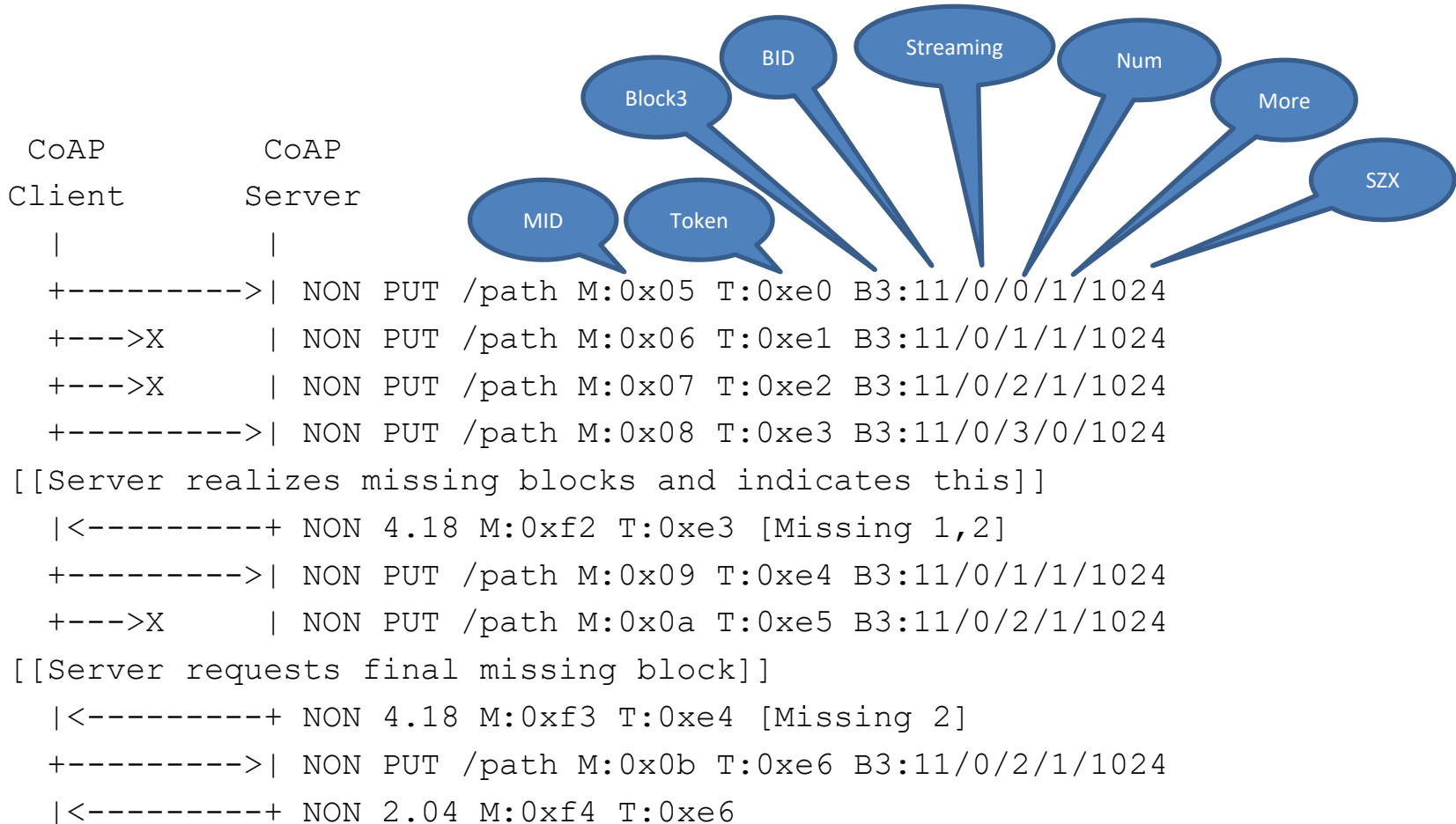
# Block3 Updates (1 of 2)

- 'A' Bit removed. That bit has now a new meaning : 'S' streaming bit
  - 'S' bit set when sending streamed data that has no size
- Block ID (BID)
  - Need to differentiate between different PUTs (with different body) to same resource
    - To handle failure conditions
  - Unable to use Token here as alternative as each request must have a different token
  - Cannot have a value of 0 (Block4 special case)
- Tokens must not be empty
  - Each PUT has different Token
  - Error responses for particular block can be handled

# Block3 Updates (2 of 2)

- Partial body clean up comments
- 4.08 (Request Entity Not Complete) not recommended
  - Blocks may arrive out of order (no longer “lock stepping”)
- New TBA3 (4.18) (Missing Payloads) instead of 4.08
  - Indicates missing blocks in response payload
    - CBOR encoded – count + list of missing block numbers
- 2.31 (Continue) not used

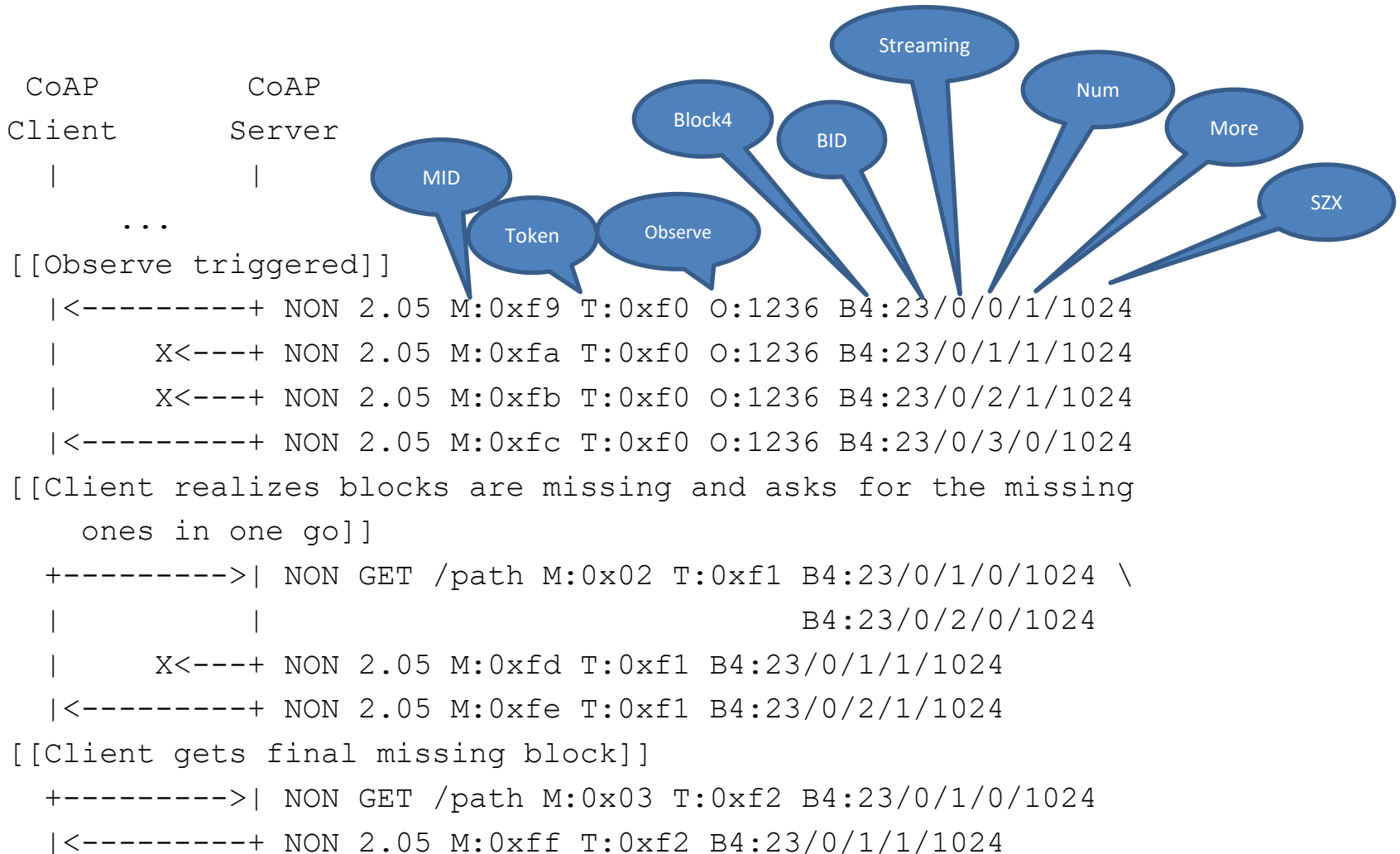
# Block3: An Example



# Block4 Updates

- 'A' Bit removed. That bit has a now a new meaning : 'S' streaming bit
  - 'S' bit set when receiving streamed data that has no size
- Block ID (BID)
  - Needed to indicate which "body" is missing some blocks
  - Cannot use ETag as an alternative here
    - 2.03 (Valid) response usage conflicts here
  - Each "body" response must have a non zero BID value
    - Random initial value – MUST be different per "body"
  - Can only have a value of 0 when requesting all blocks of (new) "body"

# Block4 Example





# Streaming Support (C→S)

- If the Size1 Option is not specified, there is support for streaming data to the server by always setting the S and M bits in the payload
  - The server may indicate missing blocks with the TBA3 (Missing Payloads) Response Code
    - This is NOT RECOMMENDED
  - The client indicates the end of the streaming of data by unsetting the S bit

# Streaming Support (S→C)

- If the Size2 Option is not specified, there is support for streaming data to the client by always setting the S and M bits in the payload
  - The client can request missing blocks
    - This is NOT RECOMMENDED
  - The server indicates the end of the streaming of data by unsetting the S bit
  - The client no longer wanting to receive data can send a GET request to the same resource with the same BID, but both M and S bits unset

# Next Steps

- Further discussion
- Consider adopting as a WG Document

# Thank You