DOTS Gateways: Options for Handling Collisions

M. Boucadair (Orange)

December, 2017

DOTS Gateways: A Reminder

- A DOTS gateway is logically equivalent to a DOTS server back-to-back with a DOTS client
  - The server and the client behave exactly like their regular counterparts
  - Some “glue” between the two is needed
    - This “glue” is currently undefined
    - ..or it is very very minimalist
- DOTS gateways are not a mandatory “piece” of the architecture…
  - …but we all know that proxies/relays/gateways end by being involved

DOTS Gateways Flavors

- Client-side gateways
  - Deployed at the client domain to intercept and process messages from internal DOTS clients
  - DOTS clients are provided with the reachability information of the GW
  - Whether a GW or “pure” DOTS clients are deployed at the client domain is opaque to the DOTS server
  - Adding/removing DOTS clients within a domain is a local engineering decision…that should be hidden from outside
  - Cascaded GWs may be considered in large networks
  - The client side of the GW can request mitigations on its own
- Server-side gateways
  - Deployed at the server domain
  - The ultimate server may need some information about the source client domain (DOTS client) for policy enforcement
  - The GW can supply some information
- Client- and server sides GWs may be simultaneously invoked on-path

Hop-by-Hop Authentication

- The Client authenticates to the DOTS GW & the DOTS GW authenticates to the next DOTS server
- This mode allows DOTS GWs to alter DOTS messages, if required.

Server-side GW: Assist the Server to enforce policies

- Supply a client (domain) identifier would be helpful
- The server is configured with trusted GWs

A Client-side Gateway May Rewrite Messages it Relays

- The typical scenario is when the GW is co-located with a translator or if it can interact with a translator (RFC 6736, RFC 8045)
- The GW may replace internal IP address and/or internal port with the external ones
- It may rewrite the heartbeat interval for optimized load

The Client-Side Gateway Maintains a State Table

- Because the GW is a DOTS server!
- It may preserve the same mitigation-id in both legs or not.
- Conflicts can be detected and handled locally

A Client-side Gateway May Trigger Updates on behalf of Clients

- For example, when the CPE of an enterprise network acquires a new IP address/prefix
  - Stale filtering rules need to be removed
  - New ones installed without waiting for the client to renew
  - Clients may not be aware of the address/prefix change (renumbering events)
    - How to handle address/prefix change by clients?

Client-Side Gateway: Handling Collisions

- Collisions may happen and can be detected by a client-side GW
  - If the request is validated, the request can be rewritten to uniquely identify the request
    - A new mitigation-id is used
    - Internal mitigation-id/external mitigation-id is stored locally
    - No impact on the packet size
- Because cascaded GWs may be deployed in a domain, rewriting mitigation-id does not reveal the internal topology at the client side
  - Does not increase the size of the packet
- All packets will expose the same format (with or without GW)

Option 1

Client-Side Gateway: Handling Collisions

- Consider that DOTS clients includes a “Request Nonce” that is randomly generated by the DOTS client
- Collisions are unlikely
  - The client-side GW relies on the mitigation-id an request nonce to identify requests
  - The client-side GW preserves the “Request Nonce” upstream
  - No metadata insertion by the GW
- All packets from DOTS clients with or without gateways will expose the same format/attributes

Option 2

Client-Side Gateway: Handling Collisions

- Mitigation-id is redefined to be a random 64-bit number
- Collisions are unlikely
  - The client-side GW relies on the mitigation-id to identify requests
  - The client-side GW preserves the mitigation-id
  - No metadata insertion by the GW
- All packets from DOTS clients with or without gateways will expose the same format/attributes

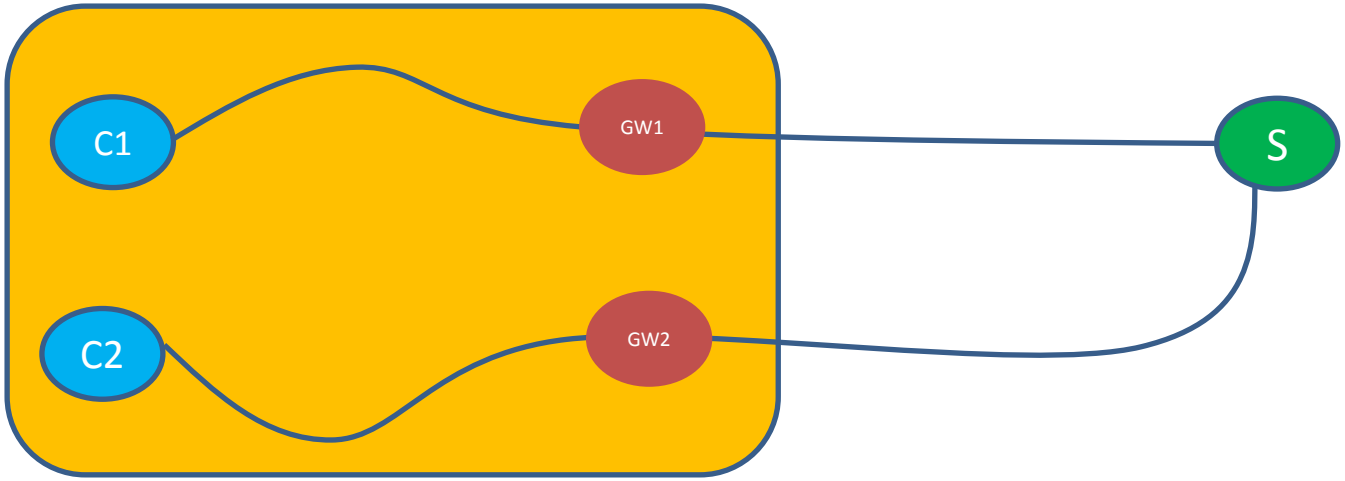
Option 3

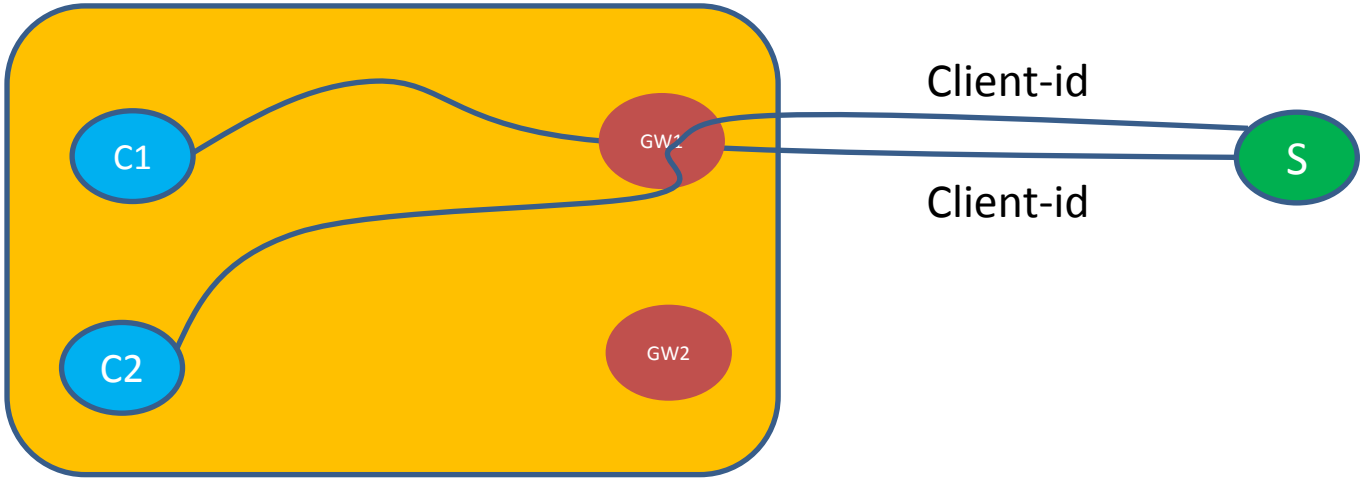
Client-Side Gateway: Handling Collisions

- Collisions may happen and can be detected by a client-side GW
  - The GW injects a header to identify the client
  - Stores the identifier and the mitigation-id
- Collisions may not be detected if the requests are handled by distinct GWs of the same domain
- Service degradation may be observed, e.g.,
  - Anycast addressing mode for the GWs
  - Subsequent requests may not be handled by the same GW…which will lead to failures because the server will reject it because the request does not match the 'client-id' it stores
- Server-side GWs must preserve the order
  - Otherwise, problems will be observed at the server side
  - There is no such issue if only the server-side GW supplies such information

Option 4

Client-Side Gateway: Handling Collisions





Option 4