Constrained RESTful Environments WG (core)

Chairs:

Andrew McGregor <andrewmcgr@gmail.com>

Carsten Bormann <cabo@tzi.org>

Mailing List:

core@ietf.org

Jabber:

core@jabber.ietf.org

- **We assume people have read the drafts**
- **Meetings serve to advance difficult issues by making good use of face-to-face communications**
- **Note Well: Be aware of the IPR principles, according to RFC 3979 and its updates**

✓Blue sheets
✓Scribe(s):
http://tools.ietf.org/wg/core/minutes

Note Well

This summary is only meant to point you in the right direction, and doesn't have all the nuances. The IETF's IPR Policy is set forth in BCP 79; please read it carefully.

The brief summary:

❖**By participating with the IETF, you agree to follow IETF processes.**

❖**If you are aware that a contribution of yours (something you write, say, or discuss in any IETF context) is covered by patents or patent applications, you need to disclose that fact.**

❖**You understand that meetings might be recorded, broadcast, and publicly archived.**

For further information, talk to a chair, ask an Area Director, or review the following:

BCP 9 (on the Internet Standards Process)

BCP 25 (on the Working Group processes)

BCP 78 (on the IETF Trust)

BCP 79 (on Intellectual Property Rights in the IETF)

Milestones (from WG charter page)

http://datatracker.ietf.org/wg/core/charter/

Document submissions to IESG:

Done	CoAP protocol specification with mapping to HTTP Rest API to IESG
Oct 2013	Blockwise transfers in CoAP to IESG
Oct 2013	Observing Resources in CoAP to IESG
Oct 2013	Group Communication for CoAP to IESG
Jan 2014	BP for HTTP-CoAP Mapping Impl to IESG
Jan 2014	CoRE Link Collections in JSON to IESG
May 2014	CoRE Interfaces to IESG
Dec 2099	HOLD (date TBD) Constrained security bootstrapping specification to IESG

draft-ietf-core-coap-18 ✓

- **Was approved 2013-07-11**
- **Still in RFC editor queue, waiting for one MISSREF:**
- **draft-mcgrew-tls-aes-ccm-ecc**
    - **Defines the DTLS ciphersuites for RPK and Cert mode: TLS_ECDHE_ECDSA_WITH_AES_128_CCM_8**
    - **(alongside with TLS_PSK_WITH_AES_128_CCM_8 for PSK mode, RFC 6655)**
    - **IESG state “Approved-announcement to be sent::Point Raised – writeup needed”**
- **draft-ietf-tls-oob-pubkey (in queue since 2014-02-06)**
    - **Defines RPK for DTLS**
    - **IESG state “Waiting for Writeup” (post IETF last call)**

draft-ietf-core-groupcomm-18

- Finished WGLC ______
- Updated version 2013-12-22
- Next step: WG chairs to find time for writeup...

WG documents

- draft-ietf-core-observe — after mini-WGLC
- draft-ietf-core-block — before 2nd WGLC
- draft-ietf-core-http-mapping

- draft-ietf-core-links-json
  - done, but waiting for more implementation experience
- draft-ietf-core-resource-directory
  - charter work needed, to resume activity!
- draft-ietf-core-interfaces
  - to resume activity!

Related Work Reports

Reminder: Internet of Things Plugtest

- When?
  - March 7–9th, 2014 (**this weekend**)
- Where?
  - London (**here**)
- Who?
  - Implementers world-wide
- How Much?
  - Free!
- Tests:
  - **CoAP** (Mandatory)
  - Block, Observe (Optional)
  - OMA Lightweight M2M (Optional)
  - **DTLS** for CoAP (Optional)
  - **6LoWPAN** (Optional)

Agenda Bashing

Tuesday

All times are in time-warped UTC

- 09:00–09:10 Intro
- 09:10–10:00 -observe (KH)
- 10:00–10:15 HTTP mapping (SL)
- 10:15–10:30 No-Response (AB)
- 10:30–11:00 Alternative Transports (BS)
- 11:00–11:15 Congestion Control (CG)
- 11:15–11:30 Flextime

Between the slots

- CoRE AA proposed work going on in:
  ACE BOF, Wed 09:00–11:30
- LWIG meeting Thu 17:00–18:30
- 6Lo Wed 15:20–17:30
- 6TiSCH Thu 13:00–15:00

Friday

- 11:50–11:55 Intro
- 11:55–12:30 ACE BOF post-processing
- 12:30–12:40 Geo link-format (TF)
- 12:40–12:55 Sleep next steps (AR)
- 12:55–13:05 CoAP for management (PV)
- 13:05–13:20 Flextime

Group 1:WG docs

Observing Resources in CoAP

draft-ietf-core-observe

Klaus Hartke

IETF 89 London

Changes since IETF88

- When a resource changes its state and there is already an active transmission to a client for a previous state, then an optimized algorithm for transmission can be used. This algorithm is now OPTIONAL.

- Clients may want to cancel an observation more eagerly than is possible by just rejecting notifications. A mechanism for proactive cancellation has been added. It is REQUIRED for servers and OPTIONAL for clients.

Cancellation

- New kind of interaction – it's not a request targeted at a resource; it's not a response providing the result of a request
- Very late addition to -observe and to CoAP in general
- How to express cancellation in CoAP messages?
    - Overload the GET method in some way
    - Introduce a new protocol element to CoAP

Overloading GET

- GET method: requests the server to update the client with a current representation of the target resource {once / over a period of time}

- Cancellation: requests the server to stop updating the client with a current representation of the target resource

- Overloading the GET method to also provide cancellation leads to unwanted side-effects (e.g., the server must return a response that the client does not want; a whole chain of proxies may be forced to update their caches; ...)

- We could redefine the GET method to have completely different semantics in some cases, but then it would become very difficult to explain

New protocol element

- Pro: Lets us cleanly express what we want to say
- Con: Risky late additions to a protocol are risky
- Currently in the draft: new code **7.31 Cancel**

0,00	Empty
0.01–0.31	Requests
1.00–1.31	*Reserved*
2.00–5.31	Responses
6.00–6.31	*Reserved*
7.00–7.31	**Control**

Influencing existing requests

- **Request matching**
  - **Hard to define exact rules**
  - **Too easy to trigger accidentally**
- **Message-ID matching**
  - **Great, but only works with response in hand (reactive)**
- **Token matching**
  - **Already required for observation renewal**
    - full request matching assumed in this case
    - accidental token reuse behavior not defined otherwise
  - **Easy to define exact match**
  - **Can’t easily hit accidentally**

HTTP-CoAP Mapping I-D Status
draft-ietf-core-http-mapping-03

draft-ietf-core-http-mapping@tools.ietf.org

IETF 89, London

Changes from IETF 88

- Closed tickets #351, #363, and #364
  - Refine URI Mapping template(s)
  - Discovery
  - Security & Privacy implications of the URI Mapping scheme
- Most of the changes in one place (Section 5)
  - Self contained edits
  - Straightforward to review
  - Please, provide comments

Closed Tickets in Detail

- #351: “Security implications of default URI mapping”
- #363: “Remove coap scheme in default URI mapping”
- #364: “Add discovery of HTTP-CoAP mapping resource”

Ticket #364

- Section 5.4 describes how to discover:
  - the *base* HTTP-CoAP Proxy resource, and
  - the associated URI mapping template.
- Two new RFC 6690 attributes:
  - the `core.hc` resource type
  - the `hct` (HTTP-CoAP Template) attribute

Ticket #364

Discover the HTTP-CoAP Proxy function and arguments

On the CoAP side:

```
REQ:   GET coap://[ff02::fd]/.well-known/core?rt=core.hc

RES:   2.05 Content
       </hc>;anchor="http://p.example.com";rt="core.hc";hct="?uri={+tu}"
```

Ticket #364

Discover the HTTP-CoAP Proxy function and arguments

On the HTTP side:

```
REQ:   GET /.well-known/core?rt=core.hc HTTP/1.1
       Host: p.example.com

RES:   HTTP/1.1 200 OK
       Date: Tue, 25 Feb 2014 23:10:21 GMT
       Content-Type: application/link-format
       Content-Length: 58

      </hc>;rt="core.hc";hct="?uri={+tu}"
```

Ticket #364

De-reference a CoAP resource via the HC Proxy

```
REQ:   GET /hc?uri= coap://s.example.com/temp  HTTP/1.1
       Host: p.example.com

RES:   HTTP/1.1 200 OK
       Date: Tue, 25 Feb 2014 23:12:36 GMT
       Content-Type: text/plain
       Content-Length: 5

       22.2C
```

Ticket #363

- Section 5.2.1 has MAY text to describe the possible suppression of the scheme part in the mapped URI;
- The suppression choice is application dependent:

```
REQ:   GET /hc?uri=s.example.com/temp HTTP/1.1
       Host: p.example.com
```

Ticket #351

- Section 8.3. has text analysing security and privacy issues introduced by the proposed URI Mapping, and providing countermeasures – where possible.

Next Iteration (-04)

Tackle the following tickets:

- #365 “Media Type conversions”
- #366 “Re-base relative URI when accessed via the proposed URI Mapping”

Request the creation of the `core.hc` and `hct` entries in the CoRE Parameters Registry (RFC6690).

Group 2: “new work”

TATA CONSULTANCY SERVICES
Experience certainty.

TATA

The No-Response option for CoAP

draft-tcs-coap-no-response-option-05

CoRE WG meeting@IETF 89

Abhijan Bhattacharyya, Soma Bandyopadhyay, Arpan Pal
TCS Innovation Labs

Copyright © 2012 Tata Consultancy Services Limited

From meeting #88

- Response to ‘No-Response’ was favorable
- Important comments received in the mailing list and during the meeting
  - Re-use interval for message IDs
  - Congestion control
  - Duality with ‘observe’
  - Concerns on granularity feature

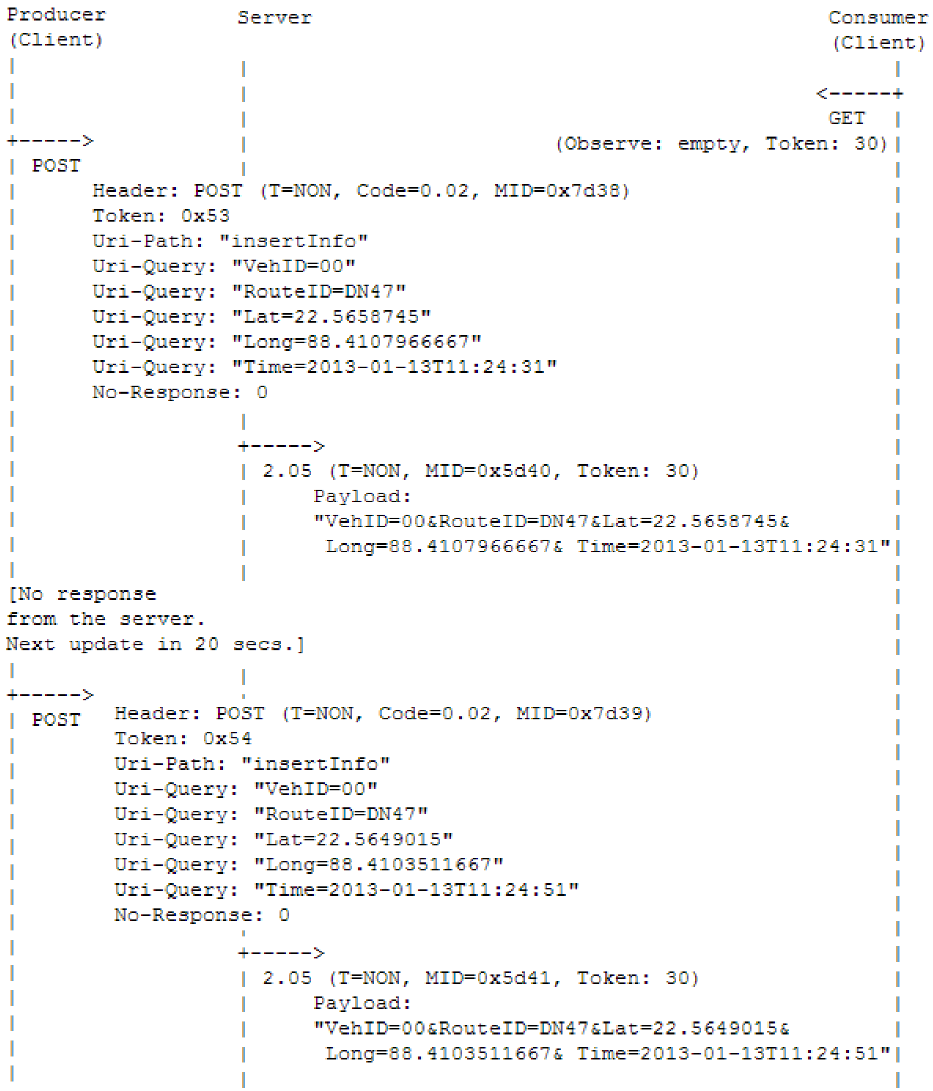
- Present version of the draft tries to address the comments – mainly section 5 of the present draft

Updates

- Re-use interval for message IDs
  - NON-LIFETIME suggested as the time interval over which a message ID can be safely re-used
- Addressing congestion
  - Possible communication scenarios primarily fall into the class of low-data volume applications (section 3.1.2 of RFC 5405 – Unicast UDP Usage Guidelines)
    - Scenario where the application cannot maintain an RTT estimate due to lack of reverse traffic
    - 3 seconds interval is 'suggested' as the minimum gap between successive updates
  - Application developer should have the flexibility to handle this
    - May interweave occasional closed-loop exchanges for time-to-time adjustments

Updates (contd.)

- Duality with ‘Observe’
  - Interest of the ‘producer’ or the ‘consumer’?
  - Just simple PUT or POST with No-Response may be quite simple in some scenarios than going through the ‘Observe’ request and response.
    - Multicasting for actuations (turning lights on, etc.) may be a good use case
  - Observe and No-Response could actually be complementing techniques for lightweight end-to-end implementations
    - Ex: Vehicle location update using No-Response and real-time tracking by a remote user using observe



```
Producer          Server                                          Consumer
(Client)                                                          (Client)
|                 |                                                     |
|                 |                                               <-----+
|                 |                                                 GET |
+----->           |                         (Observe: empty, Token: 30) |
| POST            |                                                     |
|      Header: POST (T=NON, Code=0.02, MID=0x7d38)                      |
|      Token: 0x53                                                      |
|      Uri-Path: "insertInfo"                                           |
|      Uri-Query: "VehID=00"                                            |
|      Uri-Query: "RouteID=DN47"                                        |
|      Uri-Query: "Lat=22.5658745"                                      |
|      Uri-Query: "Long=88.4107966667"                                  |
|      Uri-Query: "Time=2013-01-13T11:24:31"                            |
|      No-Response: 0                                                   |
|                 |                                                     |
|                 +----->                                               |
|                 | 2.05 (T=NON, MID=0x5d40, Token: 30)                 |
|                 |     Payload:                                        |
|                 |     "VehID=00&RouteID=DN47&Lat=22.5658745&          |
|                 |      Long=88.4107966667& Time=2013-01-13T11:24:31"  |
|                 |                                                     |
[No response                                                            |
from the server.                                                        |
Next update in 20 secs.]                                                |
|                 |                                                     |
+----->                                                                 |
| POST   Header: POST (T=NON, Code=0.02, MID=0x7d39)                    |
|        Token: 0x54                                                    |
|        Uri-Path: "insertInfo"                                         |
|        Uri-Query: "VehID=00"                                          |
|        Uri-Query: "RouteID=DN47"                                      |
|        Uri-Query: "Lat=22.5649015"                                    |
|        Uri-Query: "Long=88.4103511667"                                |
|        Uri-Query: "Time=2013-01-13T11:24:51"                          |
|        No-Response: 0                                                 |
|                 +----->                                               |
|                 | 2.05 (T=NON, MID=0x5d41, Token: 30)                 |
|                 |     Payload:                                        |
|                 |     "VehID=00&RouteID=DN47&Lat=22.5649015&          |
|                 |      Long=88.4103511667& Time=2013-01-13T11:24:51"  |
|                 |                                                     |
```

Updates (contd.)

- Concerns on granularity feature
  - Actually was a suggestion from the multicast domain in the mailing list
  - Main concern is on the uncertainty regarding whether the response got suppressed or lost – example: lost error messages vs. suppressed success messages
  - May be application specific max. waiting time
  - No decision was made on whether to keep it in the draft
- Comparative results showing the benefits in the reliability vs. throughput and bandwidth trade-off presented in Sensys 2013, Mobiquitous 2013.

What next?

CoAP Transport URIs

draft-silverajan-core-coap-alternative-transports

Bill Silverajan, Teemu Savolainen

Why is a Transport URI needed?

- We want to ensure that a CoAP Transport URI identifies both the resource and endpoint, as well as how to access it
- Traditionally, URIs do the former, not the latter. This is ok if:
  - The standard clearly define transport type
  - A naming or resolving architecture exists
  - Node is computationally powerful to ascertain
- But we cannot assume these would exist in constrained environments
  - Therefore have both locator and identifier in URI

Single vs Multiple Transports

- Transport availability can be divided into the following node categories (both for end nodes and proxies)
  - Type T0 nodes have a single transport
  - Type T1 nodes have 1 or more transports, which may be in unreachable/off states but at least 1 active transport
  - Type T2 nodes have multiple active transports
- These categories will help simplify discussions and target the kinds of CoAP transport URIs we need
  - By looking at where CoAP is useful for alternative transports, we can proceed to find the best URI format

Situating Transport Information in CoAP URI

Transport Information	Req 4.1.1	Req 4.1.2	Req 4.1.3	Req 4.1.4
Scheme	🟢	🟢	🔴	🟢
Authority	🟠	🟠	🔴	🟠
Rootless Path	🟢	🔴	🟢	🟢

From Section 4.1, draft-silverajan-core-coap-alternative-transports-04:

- Req 4.1.1: Conformance to RFC3986 syntax and algorithms
- Req 4.1.2: Preserving transport info when relative references are encountered
- Req 4.1.3: Avoiding URI aliasing with multiple transports
- Req 4.1.4: Avoiding heavy DNS reliance

CoAP Transport URIs and node types

- T0 nodes have simpler requirements for CoAP transport URI, in relation to managing resource reachability and accessibility
- T1 and T2 nodes require more thought since a single resource representation is made available over multiple transports, in relation to reducing URI aliasing issues

Advanced CoAP Congestion Control: update on draft-bormann-core-cocoa and related work in progress

August Betzler, Carles Gomez, Ilker Demirkol, Josep Paradells
Universitat Politècnica de Catalunya/Fundació i2cat

carlesgo@entel.upc.edu

Carsten Bormann – Universität Bremen TZI

cabo@tzi.org

Acknowledgment: this work has been partly supported by the Spanish Government through project TEC2012-32531

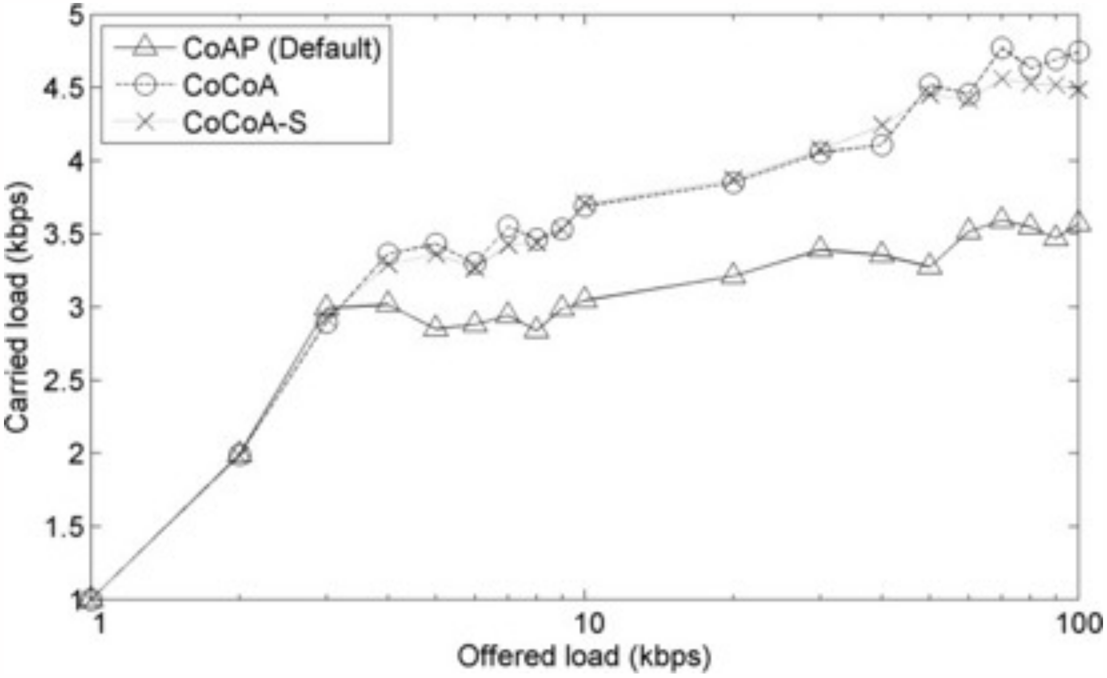
IETF 89 – London, March 2014

Context (I)

- CoAP congestion control for CONs
  - RTO chosen from a fixed interval
  - Binary Exponential Backoff (BEB)
- IETF 84 - Vancouver
  - Simple mechanism for advanced congestion control
    - Including RTT measurements
- CoCoA-00
  - draft-bormann-core-cocoa-00
  - RTO adaptively obtained based on RTT measurements
    - *Strong* and *weak* RTTs
    - RFC 6298
    - $RTO_{overall} = 0.5*RTO_{recent} + 0.5*RTO_{overall\_old}$

Context (II)

- Simulation results presented in Berlin (IETF 87)
  - Good performance of CoCoA-00



- However...

Betzler et al., “Congestion Control in Reliable CoAP Communication“, MSWIM 2013

… Underperformance of CoCoA-00

- Lower PDR in absence of MAC layer reliability
  - Greater contribution of weak RTO estimator
- Higher settling time (and delay) after a burst of transmissions
  - Large RTO increase due to a large variance of weak RTT measurements
    - E.g. average RTTVAR of 8 s in a 6x6 grid topology of 802.15.4 nodes

Changes in draft-bormann-core-cocoa-01

- RTT variance multiplier, K
  - K=4 in TCP [RFC 6298]
    - And in cocoa-00
  - Changed to K=1 for the weak estimator
    - In cocoa-01
- Rationale
  - Avoid a large RTO increase due to a large variance of weak RTT measurements

Changes in draft-bormann-core-cocoa-01

- Binary exponential backoff (BEB) for RTOs
  - CoAP base spec
  - Substituted by a Variable Backoff Factor (VBF) in cocoa-01
    - RTO < 1 s → VBF = 3
    - 1 < RTO < 8 s → VBF = 2
    - RTO > 8 s → VBF = 1.3
- Rationale
  - Avoid a sender with a small initial RTO to perform all its retransmissions in a short time interval
    - May further contribute to congestion
  - Avoid unreasonably large backoff duration values
    - Reducing maximum latency
    - Avoid blocking other messages to the same destination

Further optimizations (for -02?)

- 1) Different weighting for RTO_{weak} contribution:
  - $RTO_{overall}(i) = 0.25*RTO_{weak} + 0.75*RTO_{overall}(i-1)$
  - $RTO_{overall}(i) = 0.5*RTO_{strong} + 0.5*RTO_{overall}(i-1)$

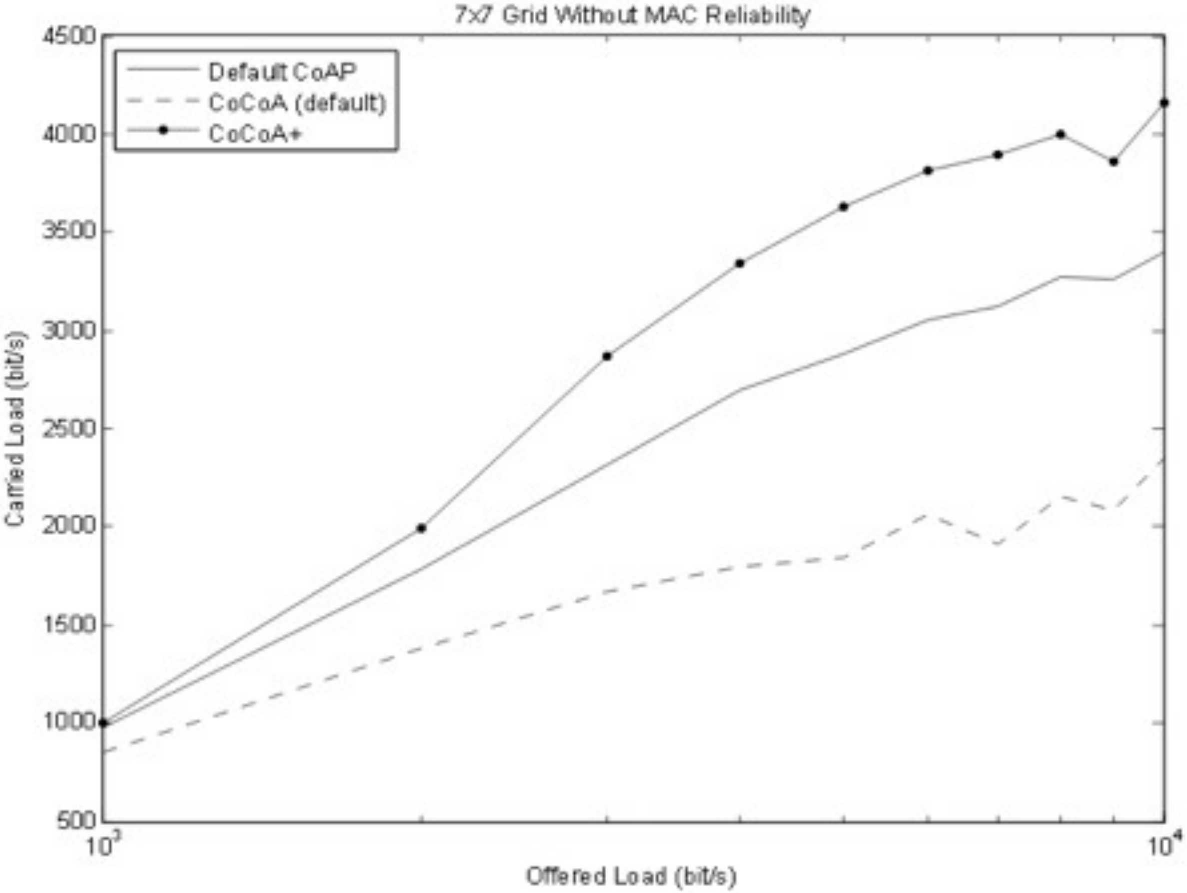
- 2) After the 2nd retransmission, the RTO is not updated

- 3) RTO aging for RTO > 2 s (not updated in last 30 s)
  - $RTO_{new} = 0.5*(2+RTO_{old})$ s
    - Closer to the default value of 2 s

- CoCoA-01 + these three optimizations = CoCoA+

CoCoA+ preliminary evaluation results (I)

- Periodic traffic
  - – Carried load Vs offered load
    - Similar to or better than default CoAP (and CoCoA-00)



CoCoA+ preliminary evaluation results (II)

- Bursty traffic
  - Better than both default CoAP and CoCoA-00
    - Up to 52% delay decrease
    - Up to 20% PDR increase
      - Vs. Default CoAP
  - CoCoA-00 decreases performance
    - Up to 30% delay increase
    - Up to 15% PDR decrease
      - Vs. Default CoAP

Conclusion and next steps

- CoCoA is being reshaped based on evaluation experience
  - Aim: wide range of conditions
    - Simulation of 802.15.4 networks
    - Topologies: grid (various sizes), chain, dumbbell
    - Traffic pattern, traffic loads
    - MAC layer reliability
  - … and performance parameters
    - PDR, delay, settling time

- WG feedback ?

Flextime

- **We assume people have read the drafts**
- **Meetings serve to advance difficult issues by making good use of face-to-face communications**
- **Note Well: Be aware of the IPR principles, according to RFC 3979 and its updates**

✓Blue sheets
✓Scribe(s)

Note Well

This summary is only meant to point you in the right direction, and doesn't have all the nuances. The IETF's IPR Policy is set forth in BCP 79; please read it carefully.

The brief summary:

❖**By participating with the IETF, you agree to follow IETF processes.**

❖**If you are aware that a contribution of yours (something you write, say, or discuss in any IETF context) is covered by patents or patent applications, you need to disclose that fact.**

❖**You understand that meetings might be recorded, broadcast, and publicly archived.**

For further information, talk to a chair, ask an Area Director, or review the following:

BCP 9 (on the Internet Standards Process)

BCP 25 (on the Working Group processes)

BCP 78 (on the IETF Trust)

BCP 79 (on Intellectual Property Rights in the IETF)

Friday

- 11:50–11:55 Intro
- 11:55–12:10 CoAP for management part deux
- 12:10–12:25 ACE BOF post-processing
- 12:25–13:05 -observe part deux
- 13:05–13:15 alternate transport URIs part deux

- ~~12:30–12:40 Geo link-format (TF)~~ Tuesday
- 13:05–13:20 Sleep next steps (AR)
- 13:05–13:20 Flextime

Authorization:
ACE BOF Post

“new work” (continued)

A Link-Format Attribute for Locating Things

draft-fossati-core-geo-link-format-attribute-03

IETF 89, London

What is geo?

- Associate geographical information to CoAP nodes/resources
- Cheap extension to CoRE Link-Format
- Re-uses geo URI syntax (RFC 5870)

Use Cases

- Rescue systems
- Gas pipeline monitoring
- Fire or flood detection
- Smart cities

Questions that geo answers

- Sensor, where are you?
- What sensors are alive in a given area?

Sensor, where are you?

Match a link-value that has a target attribute named geo:

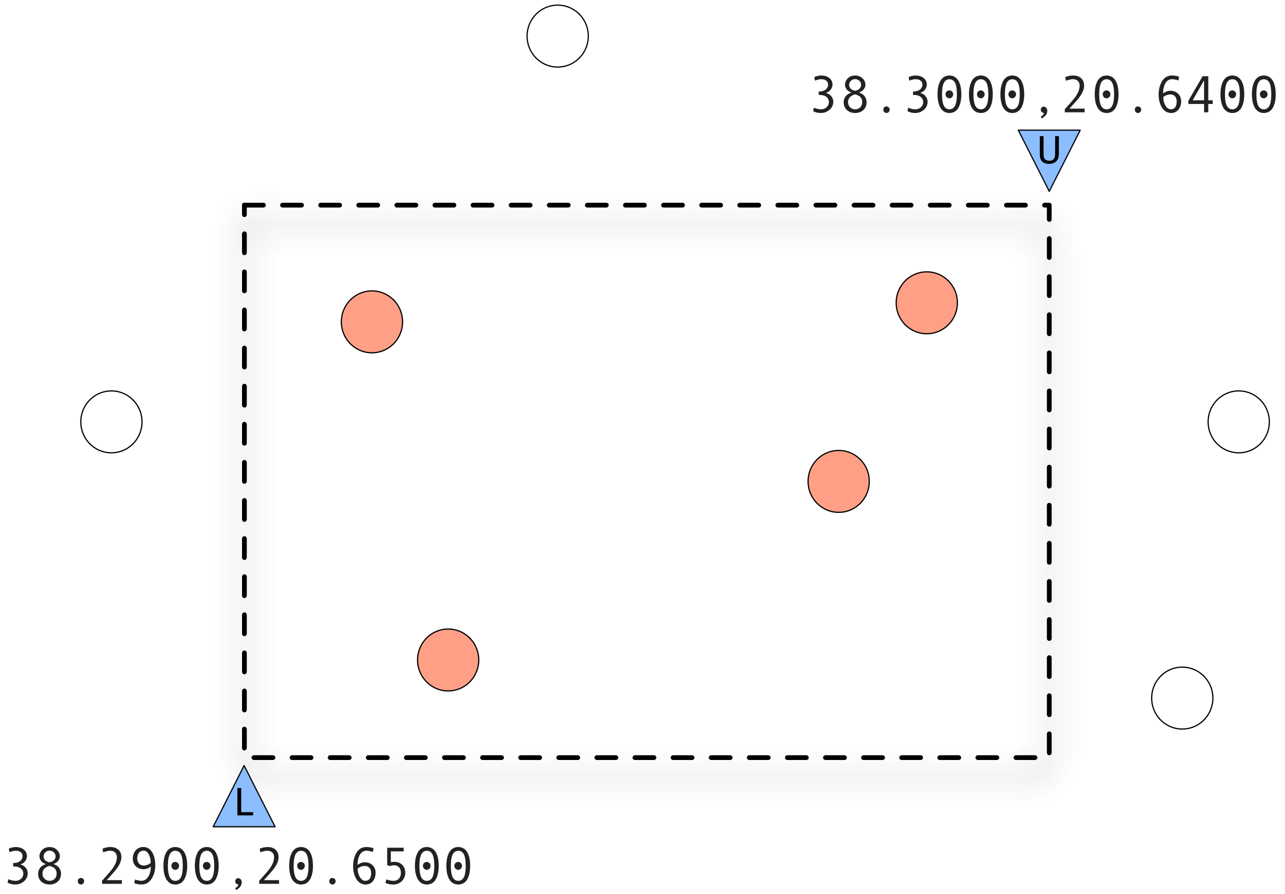
```
REQ: GET /.well-known/core?geo=*

RES: 2.05 Content
</my-loc>;geo="38.2953,20.6426,-20;u=10"
```

No code changes needed.

What sensors are alive in a given area?

Re-uses WGS84 Bounding Box from OGC



What sensors are alive in a given area? (Cont.)

Multicast query using the *extended geo query* format:

```
REQ: GET /.well-known/core?xgeo=38.3000,20.6400-38.2900,20.6500

RES[0]: 2.05 Content
</my-loc>;geo="38.2953,20.6426,-20;u=10"

RES[1]: 2.05 Content
</geo>;geo="38.2809,20.6450"

RES[2]: ...
```

Need slight code mods to parse the extended query and run the (trivial) match algorithm.

Wrap Up

- Probably useful in many contexts
- Simple and almost completely fleshed out
- Is CoRE WG interested in picking this item as an extension to RFC6690?

Sleepy Devices: Do we need to Support them in CORE?

Akbar Rahman

IETF 89, March 2014

http://tools.ietf.org/id/draft-rahman-core-sleepy-nodes-do-we-need-01.txt

Introduction

- At previous IETFs, it was suggested to review/summarize the CORE WG interest on the topic of Sleepy Node support
- Specifically whether the WG feels that support of sleepy endpoints is required for the CoAP protocol, CORE Link Format, CORE Resource Directory, etc.
- Alternatively, whether the WG feels that Sleepy Node support can be completely done outside CORE such as in the lower Layer 2 (MAC) scheduling and/or in Layer 7 (application) logic

CORE I-Ds Related to Sleepy Nodes (1/3)

- There have been multiple drafts in the CORE WG related to the subject of Sleepy Nodes including:
  - [I-D.rahman-core-sleepy-problem-statement] summarizes the overall problem space of Sleepy Nodes
  - [I-D.cao-core-aol-req] defines requirements for Sleepy Nodes to behave as if they are "always on“
  - [I-D.dijk-core-sleepy-reqs] defines requirements for Sleepy Nodes based on home and building control use cases
  - [I-D.rahman-core-sleeping] defines general requirements for Sleepy Nodes
  - [I-D.bormann-core-roadmap] provides a classification and overview of CORE drafts (and features) including a section on Sleepy Nodes

CORE I-Ds Related to Sleepy Nodes (2/3)

- [I-D.arkko-core-sleepy-sensors] describes a sensor network implementation and shows how different communication models affect implementation complexity and energy consumption (including Sleepy Node support)

- [I-D.giacomin-core-sleepy-option] defines a proxy that acts as a store-and-forward agent for a Sleepy Node

- [I-D.castellani-core-alive] defines a new CoAP message type which the Sleepy Node multicasts to all interested devices when it wakes up

- [I-D.fossati-core-publish-option] allows an endpoint to temporarily delegate authority of its resources (when it is sleeping) to a proxy server that is always on

- [I-D.fossati-core-monitor-option] extends the Observe functionality to handle the scenario when both the server and clients are Sleepy Nodes

CORE I-Ds Related to Sleepy Nodes (3/3)

- [I-D.dijk-core-sleepy-solutions] defines an architectural approach to support Sleepy Nodes
- [I-D.rahman-core-sleepy] defines new parameters that describe an endpoint's sleepy characteristics and stores them in the Resource Directory
- [I-D.vial-core-mirror-server] defines a special type of Resource Directory from which endpoints can fetch the resource regardless of the (sleep) state of the server

WG Email List Poll for Sleepy Node Deliverable

- A pulse was taken on the WG Email list asking for interest in a “CORE Sleepy Node support” deliverable

- The interesting (but non-normative) results were as follows:
  - Support FOR a new CORE Sleepy Node support deliverable: 11
  - Support AGAINST a new CORE Sleepy Node support deliverable: 3

Summary

- There have been over ten drafts related to the concept of CORE support of Sleepy Nodes

- The WG Email list pulse check on the topic showed that there was good interest in the topic of Sleepy Nodes
  - However there were some important and high profile dissenters that argued against having such a topic in CORE

- Another point to consider is that during WG discussions, the CORE Mirror Server [I-D.vial-core-mirror-server] is sometimes referred to as the "existing" solution for CORE Sleepy Node support
  - However, this draft was never adopted as a WG draft

Next Steps

- Does the WG want to keep discussing Sleepy Nodes in CORE?
  - CoAP Protocol
  - CORE Link Format
  - CORE Resource Directory
  - Etc.

- Sub-Question:
  - Does the WG want to see some immediate development of the Mirror Server concept (which seems to have near unanimous support)?

CoRE working group

draft-vanderstok-core-comi-03

CoAP Management Interfaces

Peter van der Stok, Bert Greevenbosch

March 7,2014

Overview

- CoMI defines a lightweight alternative to SNMP.
  - It wants to minimize system requirements for network management.
  - It removes the need to support both SNMP and CoAP at the same (constrained) device.
- It maintains SNMP's SMI and therefore MIBs.
- It defines new messages for reading and handling MIB variables.
- It consists of simple messages on top of CoAP.

Rationale

- Re-use of MIB definitions:
  - There is a large MIB installed base.
  - CoMI can leverage this invested MIB effort.
- Alternative to SNMP:
  - SNMP provides many aspects similar to CoAP.
  - Request / response.
  - Preferably, one packet requests and responses.
  - Security.
- Wish exists:
  - RESTful interface to MIB to reduce application development effort.
  - Integrate SNMP + CoAP to reduce code complexity and stack size.

Other work

RESTful interface to MIB and Netconf:

- [I-D.bierman-netconf-restconf] describes a restful interface to NETCONF data stores
- [RFC4088] describes a URI scheme to refer to a specific MIB instance.
- [noms2014] Carsten describes alternative URI’s

Differences between CoMI and SNMP

- Similar to YANG, CoMI uses descriptors rather than object identifiers (OIDs).
- Link between descriptors and OIDs is made in the CoAP link format.
- Tables are sent completely
  - For constrained devices, small tables are most appropriate.

Data format

- CoMI will use CBOR and JSON instead of ASN.1/BER.
- Investigating conversion :
  - MIB -> YANG -> JSON or MIB -> JSON
  - MIB -> YANG -> CBOR or MIB -> CBOR

Protocol features

- CoAP GET to read a variable.
- CoAP PUT to write a variable.
- CoAP Observe to monitor a variable.
- Filtering of table entries.
- CoAP Block to transfer payload data that is too large.
- Descriptor of the MIB variable is in the URI.

URI-query attributes

- "mod" uri-query attribute specifies the enveloping modules.
- "con" uri-query attributes specifies the context
- "row" uri-query attribute specifies the row number in a MIB table

Query key words

- _indexMIB is used to specify the index value.
- _multiMIB is used to specify an array of MIBs

CBOR encoding

- Same data and structure as JSON.
- Binary - more efficient.
- Adds concept of "translation table"
  - Instead of directly defining a descriptor-value mapping, we can use a descriptor-id and subsequent id-value mapping.
  - "id" is an integer.
  - This allows more efficient encoding, especially of recurrent fields.
  - Translation table and data can be obtained separately.

Example of CoMI Payload (JSON)

```
REQ: GET example.com/mg/mib/sysUpTime


RES: 2.05 Content (Content-Format: application/json)
{
 "sysUpTime" : 123456
}
```

Example of CoMI Payload (Table)

```
REQ: GET example.com/mg/mib/ipNetToMediaTable

RES: 2.05 Content (Content-Format: application/xxxx)
{
 "ipNetToMediaTable " : [
  {
   "ipNetToMediaIfIndex" : 1.
   "ipNetToMediaPhysAddress" : "00:00::10:01:23:45",
   "ipNetToMediaNetAddress" : "10.0.0.51",
   "ipNetToMediaType" : "static"
  },
  {
   "ipNetToMediaIfIndex " : 1,
   "ipNetToMediaPhysAddress " : "00:00::10:54:32:10",
   "ipNetToMediaNetAddress" : "9.2.3.4",
   "ipNetToMediaType " : "dynamic"
  }
 ]
}
```

Example of requesting multiple variables

```
REQ: GET example.com/mg/mib (Content-Format: application/json)
{ “_MultiMIB" : [
  { "sysUpTime" : "null"},
  { "ipNetToMediaTable" : "null" }
] }

RES: 2.05 Content (Content-Format: application/json)
{ "MultiMIB" : [
  { "sysUpTime" : 123456},
  { "ipNetToMediaTable" : [
   {
    "ipNetToMediaIfIndex" : 1,
            and other entries
   },
   {
    "ipNetToMediaIfIndex" : 1,
          and other entries
   }
  ] }
] }
```

Example of filtering table entries

```
REQ: GET example.com/mg/mib/ipNetToMediaTable(Content-Format:
   application/json)
{ “_indexMIB" :
 {
  "ipNetToMediaIfIndex" : 1,
  "ipNetToMediaNetAddress" : "9.2.3.4"
 }
}


RES: 2.05 Content (Content-Format: application/json)
{
 "ipNetTOMediaTable" : [
   {
    "ipNetToMediaIfIndex" : 1,
    "ipNetToMediaPhysAddress" : "00:00::10:01:23:45",
    "ipNetToMediaNetAddress" : "9.2.3.4",
    "ipNetToMediaType" : "static"
   }
  ]
}
```

Example CBOR encoding

JSON:

```
{
 "sysUpTime" : 123456
}
```

CBOR:

data:

```
82                           # array
   19 04d2                   # translation table ID (1234)
   a1                        # map
      00                     # 00: "sysUpTime"
      1a 0001e240            # unsigned(123456)
```

translation table:

```
82                           # array
   19 04d2                   # translation table ID (1234)
   a1                        # map
      00                     # 00
      69  737973557054696d65 # "sysUpTime"
```

Conclusion

- CoMI encourages reuse of MIB.
- Uses general (CoAP) REST methods to read and modify MIB variables.
- Removes need to support SNMP on constrained devices.
- Supports tables, as well as filtering.
- Supports JSON and CBOR.

Flextime

New slides for Friday

Constrained Management discussion

- Coman work (requirements, use cases) is completing in OPSAWG
- IETF is moving from MIB-2 to NETCONF for configuration (see recent IESG statement)
- MIB-2 is optimized for small devices [Kuryla2011, Sehgal2012]
- Proposals are being fleshed out for transporting MIB-2 information over CoAP (comi, [Bergmann2014])
- Is that a good 80 % approach?

Authentication and Authorization for Constrained Environments (ACE)

BOF Summary

Wed 09:00-11:30, Balmoral

BOF Chairs: Kepeng Li, Hannes Tschofenig

Responsible AD: Barry Leiba

Mailing List: ace@ietf.org

ACE BOF, IETF-89 London

Executive Summary

- ~ 120 participants at the BOF
- Audience expressed strong interest to work on this topic.
- ~ 30 persons with interest to review specifications.
- ~ 15 persons with interest to write specifications.
- High level summary posted to the mailing list.

Constrained Node Networks

- Carsten explained what constrained node networks are and why they pose challenges for protocol designers.
- Slides:
  - http://www.ietf.org/proceedings/89/slides/slides-89-ace-2.pdf

Use Cases and Requirements

- Ludwig introduced the container monitoring use case.
- Feedback:
  - Consider looking at the entire lifecycle of these IoT devices.
  - Assumptions regarding proxies, and connectivity require more discussion/analysis.
- Slides:
  - http://www.ietf.org/proceedings/89/slides/slides-89-ace-3.pdf

Architecture Design Choices

- Goeran presented the architecture design choices.
- Feedback:
  - Is it assumed that devices have a clock?
  - Does the work focus only on CoAP and DTLS or other transport bindings in focus?
  - More discussion needed regarding symmetric vs. asymmetric cryptography.
- Slides:
  - http://www.ietf.org/proceedings/89/slides/slides-89-ace-4.pptx

Gap Analysis

- Hannes presented four IETF security protocols (Kerberos, ABFAB, Oauth, and PKI) and asked the question whether they could be re-used?
- Various gaps have been identified but the final verdict depends on the agreed requirements and assumptions.
- Chairs encouraged to “play around” with strawman proposals to gain experience.
- Slides:
  - http://www.ietf.org/proceedings/89/slides/slides-89-ace-5.pptx

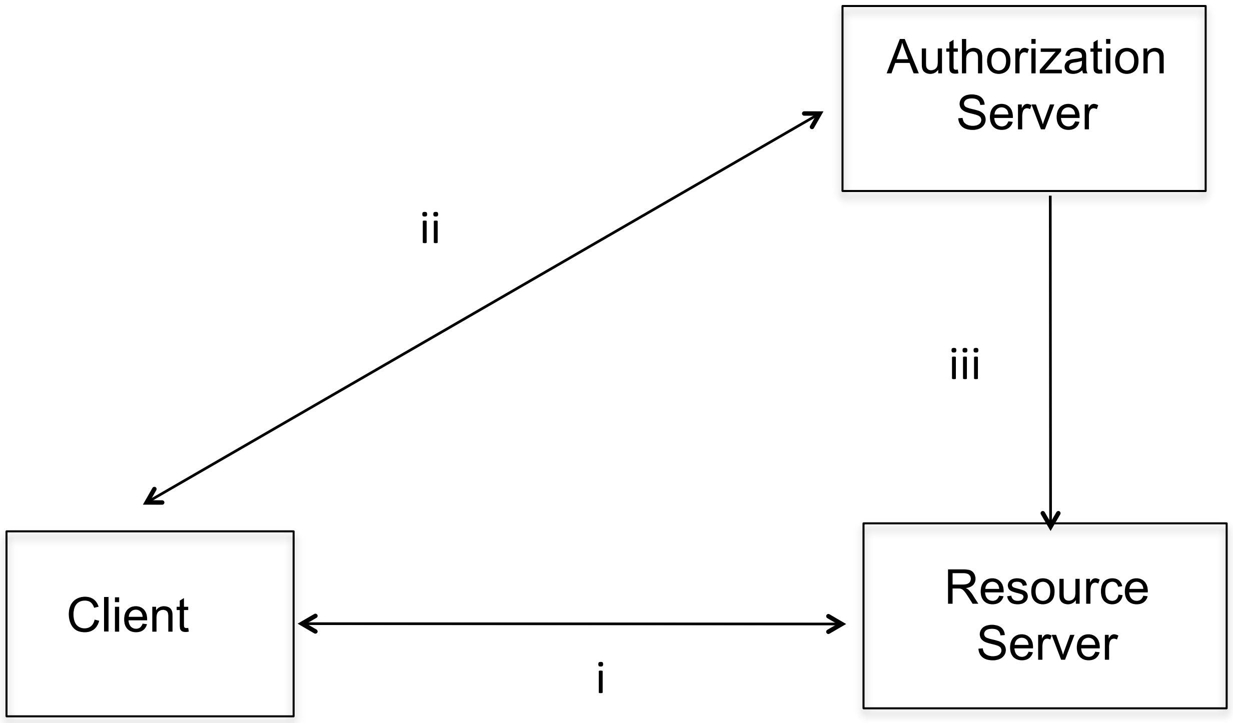
Charter Discussion

- Scope of the work needs to be clarified.
  - Is the focus only on CoAP and DTLS?
  - Chairs started mailing list discussion already.
- Milestones and task lists got simplified.
- Suggestion was made to document use cases, requirements, and design considerations on a Wiki page rather than producing RFCs.
- Solution specific charter items will be removed.

Scope Discussion

- Difficult to decide whether or not to restrict scope to CoAP & DTLS.
  - For: we can’t define general protocol working for everything.
  - Against: we should allow other transports.
- Chair’s proposal: initial focus should be use of CoAP in DTLS, but other bindings for CoAP and other transport would be possible later.
  - No consensus.

Open Issue: Scope



- What protocols should be used in which of these interfaces?

Next Step

- Discuss the scope and charter in the mailing list, and get consensus.

CoAP Transport URI formats
72 hours later at IETF 89…

draft-silverajan-core-coap-alternative-transports

Bill Silverajan, Teemu Savolainen

Situating Transport Information in CoAP URI

Transport Information	Req 4.1.1	Req 4.1.2	Req 4.1.3	Req 4.1.4
Scheme	🟢	🟢	🔴	🟢
Authority	🟠	🟠	🔴	🟠
Rootless Path	🟢	🔴	🟢	🟢

From Section 4.1, draft-silverajan-core-coap-alternative-transports-04:

- Req 4.1.1: Conformance to RFC3986 syntax and algorithms
- Req 4.1.2: Preserving transport info when relative references are encountered
- Req 4.1.3: Avoiding URI aliasing with multiple transports
- Req 4.1.4: Avoiding heavy DNS reliance

Downgrade URI Aliasing as a design requirement

Transport Information	Req 4.1.1	Req 4.1.2	Req 4.1.3	Req 4.1.4
Scheme	🟢	🟢	○	🟢
Authority	🟠	🟠	○	🟠
Rootless Path	🟢	🔴	○	🟢

Req 4.1.3: Avoiding URI aliasing with multiple transports

- Should not be a stringent requirement as in practice cannot be avoided by URI format alone
- Reduce its priority to a recommended practice

Express Alternative Transport in URI scheme

Transport Information	Req 4.1.1	Req 4.1.2	Req 4.1.3	Req 4.1.4
Scheme	●	●	○	●

- Examples:
  - coap+tcp://[2001:db8::1]:5683/sensors/temperature
  - coap+sms://0015105550101/sensors/temperature
- Next draft version looks at how T1 and T2 nodes can express alternate resource representations to clients

No URI Scheme prefix registration

- The idea of registering URI scheme prefixes as namespaces, such as “ms” to form “ms-word” as URI schemes, was voted down in appsarea wg
- Instead our draft would choose to specify recommendations and guidelines for new transport URI schemes instead of regulating it
- Delimiter character between “coap” and “transport-type” would rest on appsarea wg decision by July 2014.
  – So could be “coap+ws” or “coap.ws” or “coap-ws”…

Observe Cancellation

(Slides by Matthias Kovatsch,
with edits by Carsten Bormann)

RST with Last Notification MID

Still works for reactive cancellation

PRO

- No re-use of active Token
- No representation on cancel

CON

- Not available over alternative transports
- Issues around MID reuse

PUT/POST/DELETE with Observe Option

- It is not a GET

- Observe is elective and unsafe methods might corrupt the resource state

GET without Observe Option

PRO

- Allows for several different implementation strategies
- Already tested

CON

- Re-use of active Token
- Triggers response (maybe via intermediaries)
- Every GET (for obs resource) needs to check Observe relation list for Token

7.xx / 0.31 Control Message

PRO

- No re-use of active Token for cancellation
- No response on cancel
- Could be extended for separate responses with liveliness check

CON

- Still re-use of active Token for re-registration
- New class at late stage (cf. ping)
- Beware of too many new control messages (unRESTfulness)

Who can not live with?

- Get like renew (same token) with same request parameters and Observe=17

- Cancel message 0.31 (same token)