Constrained RESTful Environments WG (core)

Chairs:

Jaime Jiménez <jaime.jimenez@ericsson.com>

Carsten Bormann <cabo@tzi.org>

Mailing List:

core@ietf.org

Jabber:

core@jabber.ietf.org

Stand-In chair today: Francesca Palombini. Thank you!

Jaime is busy working on the supply of future working group chairs ➔

- **We assume people have read the drafts**
- **Meetings serve to advance difficult issues by making good use of face-to-face communications**
- **We work as individuals and <u>try to be nice to each other</u>**
- **Note Well: Be aware of the IPR principles, according to RFC 8179 and its updates**

★Blue sheets
★Scribe(s)

Note Well

This is a reminder of IETF policies in effect on various topics such as patents or code of conduct. It is only meant to point you in the right direction. Exceptions may apply. The IETF's patent policy and the definition of an IETF "contribution" and "participation" are set forth in BCP 79; please read it carefully.

As a reminder:

- By participating in the IETF, you agree to follow IETF processes and policies.
- If you are aware that any IETF contribution is covered by patents or patent applications that are owned or controlled by you or your sponsor, you must disclose that fact, or not participate in the discussion.
- As a participant in or attendee to any IETF activity you acknowledge that written, audio, video, and photographic records of meetings may be made public.
- Personal information that you provide to IETF will be handled in accordance with the IETF Privacy Statement.
- As a participant or attendee, you agree to work respectfully with other participants; please contact the ombudsteam (https://www.ietf.org/contact/ombudsteam/) if you have questions or concerns about this.

Definitive information is in the documents listed below and other IETF BCPs. For advice, please talk to WG chairs or ADs:

- BCP 9 (Internet Standards Process)
- BCP 25 (Working Group processes)
- BCP 25 (Anti-Harassment Procedures)
- BCP 54 (Code of Conduct)
- BCP 78 (Copyright)
- BCP 79 (Patents, Participation)
- https://www.ietf.org/privacy-policy/ (Privacy Policy)

Agenda Bashing

All times are in time-warped EDT (UTC–04:00)

Tuesday (60 min)

- **17:10–17:20 Intro, Agenda, Status**
- **17:20–17:35 RD-DNS-SD (PV)**
- **17:35–17:50 Pubsub (KH? MK?)**
- **17:50–17:55 Dynlink (KH? MK?)**
- **17:55–18:10 CoRECONF**

All times are in time-warped EDT (UTC–04:00)

Thursday (120 min)

- 10:00–10:05 Intro, Agenda
- 09:05–09:20 OSCORE groupcomm (MT)
- 09:20–09:30 OSCORE discovery (MT)
- 09:30–09:45 Observe multicast notifications (MT)
- 09:45–10:00 Groupcomm bis (ED)
- 10:00–10:10 SenML data ct (AK)
- 10:10–10:20 SenML local base name (HT)
- 10:20–10:30 SenML units (CB)
- 10:30–10:35 Fasor
- 10:35–11:00 Flextime

Hallway discussions and side meetings

- Hypermedia in CoRE [CoRAL]:
  Tuesday 15:00..17:00, Collier
  (right before CoRE WG Tuesday)

OSCORE

draft-ietf-core-object-security

→ RFC 8613

2019-07-09

Other document status

In IESG processing

- draft-ietf-core-multipart-ct-03 (revised I-D needed)
- draft-ietf-core-resource-directory-23 (publication requested)
- draft-ietf-core-senml-etch-04 (publication requested)

In WG last call processing:

- draft-ietf-core-stateless-01 (Revised I-D Needed)
- draft-ietf-core-echo-request-tag-05 (Revised I-D Needed)
- draft-ietf-core-hop-limit-04 (Checking revised I-D)

Other document status (2)

Expired, otherwise ready for WGLC:

- draft-ietf-core-dev-urn-03

In WG adoption call:

- draft-hartke-t2trg-coral-09 and draft-hartke-t2trg-ciri-03 (ends **today**)
- draft-veillette-core-yang-library-05 (no +???)
- draft-bormann-core-corr-clar (in limbo)

All times are in time-warped EDT (UTC–04:00)

Tuesday (60 min)

- **17:10–17:20 Intro, Agenda, Status**
- **17:20–17:35 RD-DNS-SD (PV)**
- **17:35–17:50 Pubsub (KH? MK?)**
- **17:50–17:55 Dynlink (KH? MK?)**
- **17:55–18:10 CoRECONF**

RD-DNS-SD

draft-ietf-core-rd-dns-sd-05

Peter van der Stok, Michael Koster, Christian Amsuess

IETF 105 - CoRE Working Group

-05 Updates to -04

❖ Section added:

- RD service is exported to DNS
- Examples provided with ‘st’, ‘ins, ‘exp’ parameters

14

‘st’ parameter

‘st‘ attribute maps directly to the <Service> part of a DNS-SD Service Instance Name.

- Value of 'st' attribute is pre-specified.
- Registered in the IANA Service Name and Transport Protocol Port Number Registry
- Conforms to the syntax defined in RFC 6335 Section 5.

15

Example

Req: GET /rd-lookup/res?exp
Res: 2.05 Content
<coap://[FDFD::1234]:5683/light/1>;
exp;st='oic-d-light';rt='oic.d.light';ins='Spot';d='sector';ep='node1'

An agent registers the following DNS-SD RRs, assuming a derived DNS zone name "office.example.com"

_oic-light._sub._coap._udp.office.example.com IN PTR
Spot._oic-d-light._sub._coap._udp.office.example.com
Spot._oic-d-light._sub._coap._udp.office.example.com. IN TXT
txtver=1;path=/light/1;rt=oic.d.light;d=sector
Spot._oic-d-light._sub._coap._udp.office.example.com. IN SRV
0 0 5683 node1.office.example.com.
node1.office.example.com. IN AAAA FDFD::1234

missing

RD to DNS (1)

Assume:

- IP address of RD is FDFD::124
- INS 505567 is attributed by Authorization Server

Req: GET coap://[FDFD::1234]/.well-known/core?exp

Res: 2.05 Content
<rd-lookup/res>;
exp;st=rd-lookup-res;rt="core.rd-lookup-res";ep=rd1
ins="505567",
<rd-lookup/ep>;
exp;st=rd-lookup-ep;rt="core.rd-lookup-ep";ep=rd1
ins="505572"

RD to DNS (2)

An agent registers the following DNS-SD RRs, assuming a derived DNS zone name ”bldg1.example.com”

```
_rd-lookup-res._sub._coap._udp.office.example.com                IN PTR
      505567._rdlookup-res._udp.bldg1.example.com
505567._rd-lookup-res._sub._coap._udp.bldg1.example.com.      IN TXT
      txtver=1;path=/rd-lookup/res;rt=core.rd-lookup-res
505567._rd-lookup-res._sub._coap._udp.bldg1.example.com.      IN SRV
      0 0 5683 office.example.com.
rd1.bldg1.example.com.                                  IN AAAA  FDFD::1234
_rd-lookup-ep._sub._coap._udp.bldg1.example.com                  IN PTR
      505567!8_rd-lookup-ep._udp.bldg1.example.com
505567._rd-lookup-ep._sub._coap._udp.bldg1.example.com.       IN TXT
      txtver=1;path=/rd-lookup/ep;rt= core.rd-lookup-ep
505567._rd-lookup-ep._sub._coap._udp.bldg1.example.com.       IN SRV
      0 0 5683 bldg1.example.com.
rd1.bldg1.example.com.                                  IN AAAA  FDFD::1234
```

TODO ?

- Explain how to derive <Domain> part of DNS-SD Service Instance Name
- Transport (_coap, _coaps) needs to be added to service in DNS

19

All times are in time-warped EDT (UTC–04:00)

Tuesday (60 min)

- **17:10–17:20 Intro, Agenda, Status**
- **17:20–17:35 RD-DNS-SD (PV)**
- **17:35–17:50 Pubsub (KH? MK?)**
- **17:50–17:55 Dynlink (KH? MK?)**
- **17:55–18:10 CoRECONF**

draft-ietf-core-coap-pubsub

IETF105

July 23, 2019

Status

- Addressing the remaining issues around topic handling
  - How to handle the empty topic problem
  - How to publish link-format documents
  - How to simplify and clarify the lifetime controls
  - Keep Pub/Sub compatible with simple REST interaction for the mirror server use case

Proposal

- Make the topic configuration information a separate resource instead of a link
  - Create and Delete uses the Configuration resource
  - Enables a clear separation of topic management from the publish/subscribe interaction
  - More RESTful interaction model, topic state can be exchanged easily
  - Opportunity for better security by splitting access control
  - https://github.com/core-wg/pubsub/blob/master/proposal.txt

Topic Configuration Resource

- CoRAL Document or Link-Format Document
- Topic creation can use the CoRAL format or simple Link-Format with sensible defaults
- Topic creation results in a Configuration Resource
- First Publish results in a Data Resource
- Topic is not discoverable until first publish *(for subscribers! Can observe filtered list of topics)*
- Subscribe returns 4.04*(?)* until published
- First publish may be included in the creation

Topic Configuration Resource

+ Create only
@ At rest only

- Contains topic metadata
  - @ Topic location *(URI)*
  - @ Creation time, last publish time
  - Topic lifetime, data lifetime
  - Description, *internationalized title?*
  - Content-Format*(s?)* accepted
  - Topic state – unpublished, published, data-timeout
  - + Hint for topic path string for data resource
  - Linked or embedded representation for first publish
  - Linked or embedded representation for "tombstone"

Topic Create

- Submit a CoRAL document to the create entry point
  - Accept a CoRAL document
  - The location of the created configuration resource is returned in the "location" header element
  - The returned document contains the topic data resource path
  - The submitted document may contain or link to a document for initial publish
- Submit a Link-Format document containing a topic path hint
  - Accept a link-format document which contains a link to the data resource

Topic Discover

- Topic Discovery may use CoRAL or Link-Format
- Topics may discovered only after first publish *(*)*
- Topic discovery has some built-in filters that can use query parameters
- Sophisticated filters use Fetch with CoRAL documents
- Discovery may accept CoRAL or Link-Format documents
- Result contains a list of Links, or Configuration Representations if accepting CoRAL

Lifetime

- Topic Lifetime and Data Lifetime can be set on topic creation using CoRAL format
- Data Lifetime requires a "Tombstone" document to be available to send to subscribers upon data timeout
- Data read (GET) of an expired topic results in 4.04*(?)*
- Topic timeout results in topic removal and 4.04 sent to all subscribers

Topic Configuration Management

- The CoRAL document at the Configuration Resource location may be used to manage topics
- GET to obtain current configuration and state
- PATCH to update individual fields
- OBSERVE to obtain changes

Publish and Subscribe

- Data Resources support standard CRUD semantics with GET and PUT
- Publish/Write using PUT to the data resource
- Read using GET from the data resource
- Subscribe using OBSERVE of the data resource

Feature — Topic Hierarchy

- If a topic can be an entry point for Creation of topics, and can host a collection of topics, we can support a topic hierarchy
- Sub-topics could inherit certain topic configuration metadata from the parent topic
- Deleting parent topic would recursively delete sub-topics
- Read aggregation of sub-topics could be performed by the broker

Feature - Aggregation Topic

- Additional collections of topics may be created to perform "flat" aggregation of topics to enable bulk Read and Subscribe
- The CoRAL document would contain a list of sub-topics to be aggregated
- The Read and Notify responses would contain either all of the linked topics or only those that have been published to, according to a setting

Roadmap

- Get consensus on the proposal
- Review the design at CoRE Interim meetings between now and IETF106
- Drive to WGLC

All times are in time-warped EDT (UTC–04:00)

Tuesday (60 min)

- **17:10–17:20 Intro, Agenda, Status**
- **17:20–17:35 RD-DNS-SD (PV)**
- **17:35–17:50 Pubsub (KH? MK?)**
- **17:50–17:55 Dynlink (KH? MK?)**
- **17:55–18:10 CoRECONF**

draft-ietf-core-dynlink

IETF105

July 23, 2019

Status

- Issues still being discussed
  - Binding table operations
  - Link format; target attributes vs. link attributes
  - Observe attributes wrt. data consistency expectations
- Ongoing discussion about adding new attributes for data consistency across multiple observers
- Proposal for the link format and binding table

https://github.com/core-wg/dynlink/issues/16

Proposal for the binding table

- Make the Dynlink a resource
- Make the binding table a collection of binding resources
- The binding table has an entry point for creating binding resources
- Binding resources can be created using CoRAL or Link-Format documents
- Creation returns a location for the created binding

Binding Resource

- The binding resource contains a set of links with the subject being the binding resource itself
  - Source Link
  - Destination/Target Link
  - Binding Link
  - (other links for extended semantics/functionality)
- Resolves the target attribute issue by using a Binding Link that describes the binding itself

```
<coap://a/thermometer>;rel="binding-source",
<coap://b/temperature-input>;rel="binding-target",
<>;rel=self;bind-source=observe;bind-dest=put;refresh-time=30
```

Binding Table

- Is a collection of binding resources
- Has an entry point to create new bindings
- Returns location of created bindings in the response header of the create operation

```
POST /binding-collection/

Content-Format: (application/link-format)

Payload:

<coap://a/thermometer>;rel="binding-source", <coap://b/
tmperatuer-input>;rel="binding-target", <>;rel=self;bind-
source=observe;bind-dest=put;refresh-time=30

2.01 Created

Location: /binding-collection/1
```

Binding Table

- Bindings may be Deleted using CoAP Delete
- Bindings may be updated using CoRAL Patch
- Reading the Binding Table returns a list of links to the bindings currently in the Binding Table

```
GET /binding-collection/
Content-Format: (application/link-format)
2.05 Content
Payload:
</binding-collection/1>,
</binding-collection/2>,
</binding-collection/3>
```

Conditional Observe

- Different clients may have different starting conditions or different settings
- Results in a different sequence of notification values to different clients
- New conditional definitions may be needed for use cases that require coherent/consistent notification value sequence across all or some set of observers
  - Absolute-time base for observation intervals
  - Notify all observers on each notification

Roadmap

- Add the binding table changes if there is consensus to do so
- Design, review, and add new conditional notification attributes that can produce a consistent notification sequence across multiple observers
- Discuss how to support reliable notification, e.g. Series Transfer Pattern
- Roll changes in and review at CoRE Interim meetings between now and IETF106

All times are in time-warped EDT (UTC–04:00)

Tuesday (60 min)

- **17:10–17:20 Intro, Agenda, Status**
- **17:20–17:35 RD-DNS-SD (PV)**
- **17:35–17:50 Pubsub (KH? MK?)**
- **17:50–17:55 Dynlink (KH? MK?)**
- **17:55–18:10 CoRECONF**

CORECONF

Andy Bierman
Michel Veillette
Peter van der Stok
Alexander Pelov
Ivaylo Petrov

44

draft-ietf-core-sid status update

- Changes between v06 and v07
  - Removed mentions of deltas that were irrelevant
  - Updated the Terminology section with the latest template
  - ‘sid’ typedef moved from ietf-comi to ietf-sid file

draft-ietf-core-sid next step

- Should be ready for Working Group Last Call

draft-ietf-core-yang-cbor status update

- Changes between v08 and v10
  - Added more details how names can be encoded
  - Union of enums values overlapping solved
  - Union of bits values overlapping solved
  - Clarified sec 6.13.2 - names as instance-identifier and namespaces
  - More details in sec 8.1 - tag registry

draft-ietf-core-yang-cbor status update

- Upcoming changes
  - Based on a review from Carsten
  - Updated the Terminology section with the latest template
  - Clarified delta usages
  - In the examples ‘+’ will not be needed due to explicit reference SID with value 0
  - A number of editorial changes to improve readability

draft-ietf-core-yang-cbor status update

- Minor questions
  - Is examples context clear as it is right now

```
{
    1720 : {                              / system-state (SID 1720) /
        1 : {                             / clock  (SID 1721) /
            2 : "2015-10-02T14:47:24Z-05:00", / current-datetime (SID 1723) /
            1 : "2015-09-15T09:12:58Z-05:00"  / boot-datetime (SID 1722) /
        }
    }
}
```

49

draft-ietf-core-yang-cbor next step

- Should be ready for Working Group Last Call

draft-veillette-core-yang-library status update

- WG adoption call
  - Andy Bierman asked about more clarity in problem statement and proposes to simply augment existing YANG library
- Changes between v03 and v05
  - Updated the Terminology section with the latest template
  - SID references updated from CoMI to SID draft

draft-ietf-core-comi status update

- Have had interoperable implementations on previous version
- v05
  - based on reviews from ML
  - Changed complete solution name from CoMI to CORECONF
  - Examples are now based on models from existing RFCs not YANG models in appendix.
  - Added description of filter query option
  - Appendix A yang model updated not to include the SID definition as this is defined in the sid draft

draft-ietf-core-comi status update

- v06 and v07
  - v07 is the latest available since yesterday
  - based on review from Carsten
  - Updated the Terminology section with the latest template
  - sec 2.1 was split into two subsections to provide better structure
  - some possible ambiguity is removed from the description of base64 encoding of SIDs
  - SIDs vs SID deltas is explained more explicitly
  - Discovery made more clear
  - sec 8 - Security considerations - removed some text that was not providing any valuable input
  - Clarified notifications
  - Clarified SIDs that were just put as values, which was confusing
  - A number of other editorial changes to improve readability

draft-ietf-core-comi next step

- Should be ready for Working Group Last Call

Questions and answers

Thank you!

Constrained RESTful Environments WG (core)

Chairs:

Jaime Jiménez <jaime.jimenez@ericsson.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 8179 and its updates**

üBlue sheets
üScribe(s)

Note Well

This is a reminder of IETF policies in effect on various topics such as patents or code of conduct. It is only meant to point you in the right direction. Exceptions may apply. The IETF's patent policy and the definition of an IETF "contribution" and "participation" are set forth in BCP 79; please read it carefully.

As a reminder:

- By participating in the IETF, you agree to follow IETF processes and policies.
- If you are aware that any IETF contribution is covered by patents or patent applications that are owned or controlled by you or your sponsor, you must disclose that fact, or not participate in the discussion.
- As a participant in or attendee to any IETF activity you acknowledge that written, audio, video, and photographic records of meetings may be made public.
- Personal information that you provide to IETF will be handled in accordance with the IETF Privacy Statement.
- As a participant or attendee, you agree to work respectfully with other participants; please contact the ombudsteam (https://www.ietf.org/contact/ombudsteam/) if you have questions or concerns about this.

Definitive information is in the documents listed below and other IETF BCPs. For advice, please talk to WG chairs or ADs:

- BCP 9 (Internet Standards Process)
- BCP 25 (Working Group processes)
- BCP 25 (Anti-Harassment Procedures)
- BCP 54 (Code of Conduct)
- BCP 78 (Copyright)
- BCP 79 (Patents, Participation)
- https://www.ietf.org/privacy-policy/ (Privacy Policy)

All times are in time-warped EDT (UTC–04:00)

Thursday (120 min)

- 10:00–10:05 Intro, Agenda
- 09:05–09:20 OSCORE groupcomm (MT)
- 09:20–09:30 OSCORE discovery (MT)
- 09:30–09:45 Observe multicast notifications (MT)
- 09:45–10:00 Groupcomm bis (ED)
- 10:00–10:10 SenML data ct (AK)
- 10:10–10:20 SenML local base name (HT)
- 10:20–10:30 SenML units (CB)
- 10:30–10:35 Fasor
- 10:35–11:00 Flextime

All times are in time-warped EDT (UTC–04:00)

Thursday (120 min)

- 10:00–10:05 Intro, Agenda
- 09:05–09:20 OSCORE groupcomm (MT)
- 09:20–09:30 OSCORE discovery (MT)
- 09:30–09:45 Observe multicast notifications (MT)
- 09:45–10:00 Groupcomm bis (ED)
- 10:30–10:35 Fasor
- 10:00–10:10 SenML data ct (AK)
- 10:10–10:20 SenML local base name (HT)
- 10:20–10:30 SenML units (CB)
- 10:35–11:00 Flextime

10 years ago: 2009-07-28

- 2009-07-28: 6lowapp Bar BOF
  - In a meeting room at IETF75 in Stockholm
  - Actual beer (thank you, Zach Shelby and Sensinode ≫ ARM)
  - ≥ 5 area directors in the room

  - Agreement to start work on an application layer protocol that would complement 6lowpan
  - Agreement *not* to pursue separate transport layer protocol work for constrained node networks (we thought that would take us 10 years :-)

- 2009-12-24: draft-shelby-6lowapp-coap-00 (mostly requirements, though)
- 2010-03-09: CoRE WG established (met first at IETF 77, Anaheim)
- 2013-07-15: draft-ietf-core-coap-18 approved (8 yes ballots in IESG)
- 2014-06-22: RFC 7252 published

All times are in time-warped EDT (UTC–04:00)

Thursday (120 min)

- 10:00–10:05 Intro, Agenda
- 09:05–09:20 OSCORE groupcomm (MT)
- 09:20–09:30 OSCORE discovery (MT)
- 09:30–09:45 Observe multicast notifications (MT)
- 09:45–10:00 Groupcomm bis (ED)
- 10:00–10:10 SenML data ct (AK)
- 10:10–10:20 SenML local base name (HT)
- 10:20–10:30 SenML units (CB)
- 10:30–10:35 Fasor
- 10:35–11:00 Flextime

Group OSCORE - Secure Group Communication for CoAP

draft-ietf-core-oscore-groupcomm-05

Marco Tiloca, RISE
Göran Selander, Ericsson
Francesca Palombini, Ericsson
Jiye Park, Universität Duisburg-Essen

IETF 105, CoRE WG, Montreal, July 25th, 2019

Selected updates (1/3)

- Now referring *draft-dijk-core-groupcomm-bis* …
  - … that aims at obsoleting RFC 7390

- Handling replied/repeated responses on clients
  - *draft-dijk-core-groupcomm-bis* enables Observe requests over multicast
  - At most 1 fresh response from each server, except for Notifications
  - Per-request list with Recipient IDs of valid received responses
  - Delete the list when freeing up the Token value
  - TODO: handle at the CoAP layer; text in *draft-dijk-core-groupcomm-bis*

Selected updates (2/3)

- More parameters in the Common Security Context
  - "Countersignature Algorithm"
  - (Optional) "Countersignature Algorithm Parameters"
  - (Optional) "Countersignature Key Parameters" ← **NEW**

- Two different external_aad
  - Both with the countersignature parameters
  - One is for the encryption/decryption
  - One is for the signature verification
  - Now the signature covers the OSCORE option

```
aad_array = [
   oscore_version : uint,
   algorithms : [alg_aead : int / tstr ,
                 alg_countersign : int / tstr ,
                 ? par_countersign : any ,
                 ? par_countersign_key : any],
   request_kid : bstr,
   request_piv : bstr,
   options : bstr
]
```

```
aad_array = [
   oscore_version : uint,
   algorithms : [alg_aead : int / tstr ,
                 alg_countersign : int / tstr ,
                 ? par_countersign : any ,
                 ? par_countersign_key : any],
   request_kid : bstr,
   request_piv : bstr,
   OSCORE_option: bstr,
   options : bstr
]
```

Selected updates (3/3)

› Extended security considerations, mostly …

› Section 8.6 - Why the OSCORE option is now covered by the signature
  – Practically prevent a cross-group injection attack by forging the MAC
  – Many input from John and Jim

› Sections 8.7-8.14 – Kindred considerations as in OSCORE

66

Open points (1/2)

- What countersignature algorithm(s)?
  - Signature size vs. computing speed
  - ECDSA, Ed25519 (now MTI)
  - More on the Github isse #7

- Gid format as {Prefix+Epoch}
  - Now it's just an example. Should we recommend/mandate it?
  - By construction, it ensures both:
    - Uniqueness of Gids for a same GM (now a non normative "should")
    - Change of Gid value after a group rekeying
  - Conveniently used in *core-oscore-discovery* and *ace-key-groupcomm-oscore*

Open points (2/2)

› Selected from Ludwig's review – Thanks!

› Group Identifiers (Sections 1.1 and 8.5)
  – "Group Identifier (Gid): identifier assigned to the group. Group Identifiers **should be unique** within the set of groups of a given Group Manager, in order to avoid collisions."
  – No impact on security, due to different Master Secrets.
  – Mandate uniqueness under the same Group Manager? E.g. normative SHOULD?

68

› Key rotation (Section 8.4)
  – Upon rekeying, C is slow to switch to the new context
  – C protects a request with the old context, S has already the new context
  – Should we **forbid** S to shortly retain the old context? (Now left to application policies)

Summary from the Hackathon

- Three implementations went under interop tests
  - RISE (Java, Californium)
  - Peter (C, Libcoap)
  - Jim (C#, August Cellars)

- Results (more will come on the list)
  - Most of the test specification [1] got covered
  - All performed tests were successful
  - Remaining tests are on optional features

[1] *https://ericssonresearch.github.io/Multicast-OSCOAP/*

Next steps

- Close open points, e.g.:
  - Must-support countersignature algorithm(s)
  - Just-mention / recommend / should use Gid as {Prefix+Epoch}
  - Uniqueness of Group IDs
  - Short term retaining of Security Contexts

- Adopt latest input from Jim
  - Improve encoding of the external_aad
  - Clarify it's up to the GM to assert consistency of public keys

- Ready for WGLC ?

Thank you!

Comments/questions?

https://github.com/core-wg/oscore-groupcomm

All times are in time-warped EDT (UTC–04:00)

Thursday (120 min)

- 10:00–10:05 Intro, Agenda
- 09:05–09:20 OSCORE groupcomm (MT)
- 09:20–09:30 OSCORE discovery (MT)
- 09:30–09:45 Observe multicast notifications (MT)
- 09:45–10:00 Groupcomm bis (ED)
- 10:00–10:10 SenML data ct (AK)
- 10:10–10:20 SenML local base name (HT)
- 10:20–10:30 SenML units (CB)
- 10:30–10:35 Fasor
- 10:35–11:00 Flextime

Discovery of OSCORE Groups with the CoRE Resource Directory

draft-tiloca-core-oscore-discovery-03

Marco Tiloca, RISE
Christian Amsüss
Peter van der Stok

IETF 105, CoRE WG, Montreal, July 25th, 2019

Recap (1/2)

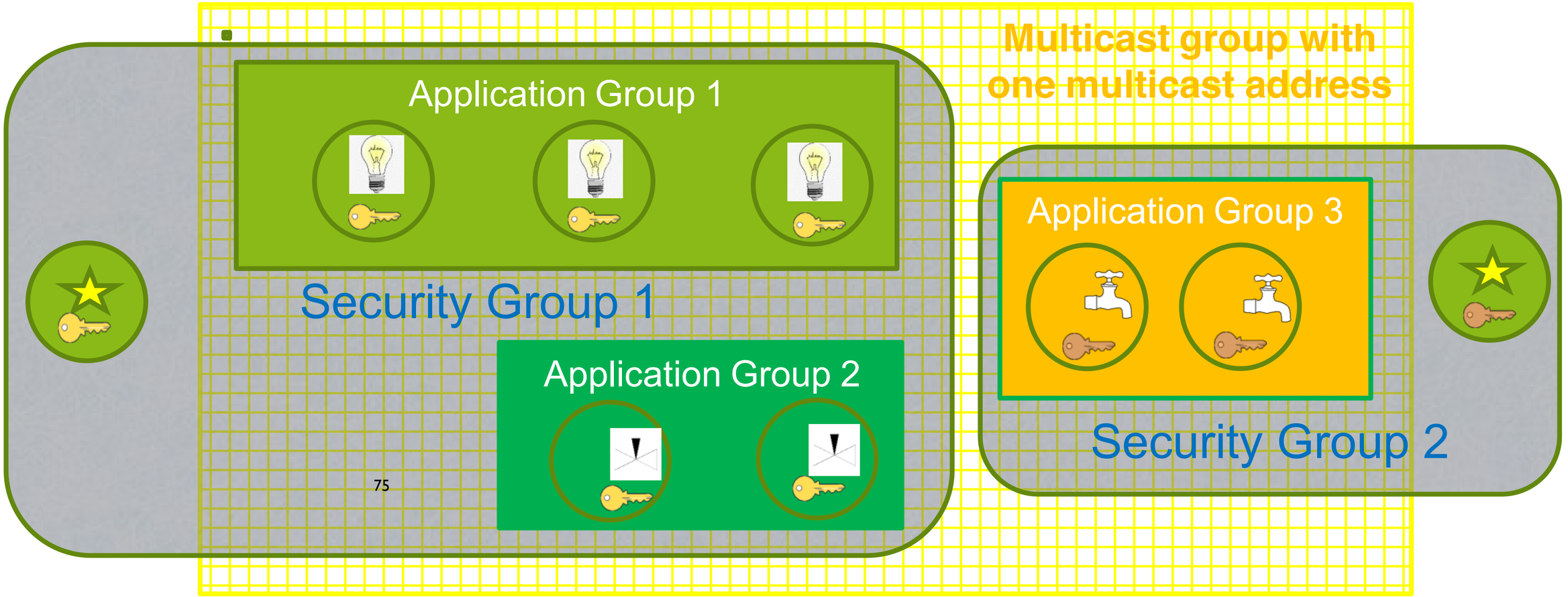
- A newly deployed device:
  - May not know the OSCORE groups and their Group Manager (GM)
  - May have to wait GMs to be deployed or OSCORE groups to be created

- Use the CoRE Resource Directory (RD):
  - Discover an OSCORE group and retrieve information to join it
  - CoAP Observe supports early discovery and changes in group information
  - Consistent with the join process in *ace-key-groupcomm-oscore*

74

- Use resource lookup, to retrieve especially:
  - A pointer to the join resource at the GM
  - The identifier of the OSCORE group

Recap (2/2)



Client of application group

Different key sets

Resources for given function

Updates from -02 (1/2)

- Full and detailed step-by-step example – Section 6
  - Aligned with a practical installation in a lighting scenario
  - Input and revision from Dave Robin (BACnet) – Thanks!

- List of steps
  1. Assign names to endpoints
  2. Register application groups to the RD
  3. Register endpoints in the application groups in the RD
  4. Register security groups, i.e. join resources of the GM, in the RD → **[This document]**
  5. Discovery of application groups
  6. Discovery of IP multicast addresses of application groups
  7. Discovery of security groups used by the application groups → **[This document]**
  8. Join the security group through the GM, e.g. as in *ace-key-groupcomm-oscore*

Initiator: Commissioning Tool

Initiator: joining node

76

Updates from -02 (2/2)

- New optional parameters for a registered join resource
  - (*)(**) *cs_alg* : countersignature algorithm, e.g. “EdDSA”
  - (*) *cs_crv* : countersignature curve (if applicable), e.g. “Ed25519”
  - (*) *cs_kty* : countersignature key type, e.g. “OKP”
  - (*) *cs_kenc* : encoding of public keys, e.g. “COSE_Key”
  - (**) *alg* : AEAD algorithm
  - (**) *hkdf* : HKDF algorithm

- Benefits for a joining node, when discovering the OSCORE group
  - (*) No need to ask the GM or to have a trial-and-error when joining the group
  - (**) Decide whether to join the group or not, based on supported the algorithms

Open point

- Parameters as link target attributes
  - Mandatory: *oscore-gid* and *app-gp*
  - Optional: *cs_alg* , *cs_crv* , *cs_kty* , *cs_enc* , *alg* , *hkdf*
  - Ideally, we should register them and where they take value from

- There is no register for link target attributes
  - Jaime started [1] to collect target attributes and discuss about this
  - Klaus has been looking into revising some parameter registries

- Right time for a new registry?

[1] https://hackmd.io/AfjWKj7rRDiQI16WSSIa4w?both

Summary and next steps

- Main updates
  - Full step-by-step example on a lighting installation scenario
  - New parameters with group information, to further facilitate joining

- Outcome from IETF 104 [2]
  - “Time to start reading it in order to decide for WGA”
  - People volunteered to review (Jim, Carsten, Bill, Klaus)

79

- Way forward
  - Process reviews and feedback as they come

[2] https://etherpad.ietf.org/p/notes-ietf-104-core?useMonospaceFont=true

Thank you!

Comments/questions?

https://gitlab.com/crimson84/draft-tiloca-core-oscore-discovery

Backup

Application & Security Groups

- Application group
  - Defined in {RD} and reused as is
  - Set of CoAP endpoints sharing a pool of resources
  - Registered and looked up just as per Appendix A of {RD}

- OSCORE Security Group
  - Set of CoAP endpoints sharing a common Group OSCORE Security Context
  - A Group Manager registers the join resources for accessing its OSCORE Groups

82

Registration

- The GM registers itself with the RD
  - MUST include all its join resources, with their link attributes
  - New ‘rt’ value “osc.j” in the CoRE Parameters registry

```
Request: GM -> RD

Req: POST coap://rd.example.com/rd?ep=gm1
Content-Format: 40
Payload:
</join/feedca570000>;ct=41;rt="core.osc.j";
oscore-gid="feedca570000";app-gp="group1"

Response: RD -> GM

Res: 2.01 Created
Location-Path: /rd/4521
```

83

Discovery (1/2)

- The device performs a resource lookup at the RD
  - Known information: name of the Application Group, i.e. “group1”
  - Need to know: OSCORE Group Identifier; Join resource @ GM; Multicast IP address
  - ‘*app-gp*’ → Name of the Application Group, acting as tie parameter in the RD

```
Request: Joining node -> RD

Req: GET coap://rd.example.com/lookup/res?rt=core.osc.j&app-gp=group1

Response: RD -> Joining node
                    84

Res: 2.05 Content
Payload:
<coap://[2001:db8::ab]/join/feedca570000>;rt="core.osc.j";
oscore-gid="feedca570000";app-gp="group1";
anchor="coap://[2001:db8::ab]"
```

Discovery (2/2)

- The device performs an endpoint lookup at the RD
  - Still need to know the Multicast IP address
  - ‘ep’ // Name of the Application Group, value from ‘*app-gp*’
  - ‘base’ // Multicast IP address used in the Application Group

```
Request: Joining node -> RD

Req: GET coap://rd.example.com/lookup/ep?et=core.rd-group&ep=group1

Response: RD -> Joining node

Res: 2.05 Content
Payload:
</rd/501>;ep="group1";et="core.rd-group";\
base="coap://[ff35:30:2001:db8::23]"
```

All times are in time-warped EDT (UTC–04:00)

Thursday (120 min)

- 10:00–10:05 Intro, Agenda
- 09:05–09:20 OSCORE groupcomm (MT)
- 09:20–09:30 OSCORE discovery (MT)
- 09:30–09:45 Observe multicast notifications (MT)
- 09:45–10:00 Groupcomm bis (ED)
- 10:00–10:10 SenML data ct (AK)
- 10:10–10:20 SenML local base name (HT)
- 10:20–10:30 SenML units (CB)
- 10:30–10:35 Fasor
- 10:35–11:00 Flextime

Observe Notifications as CoAP Multicast Responses

draft-tiloca-core-observe-multicast-notifications-00

Marco Tiloca, RISE
Rikard Höglund, RISE
Christian Amsüss
Francesca Palombini, Ericsson

IETF 105, CoRE WG, Montreal, July 25th, 2019

Motivation (1/2)

- Observe as in RFC 7641
  - One unicast registration request to a Server S
  - Notifications from S, as unicast responses

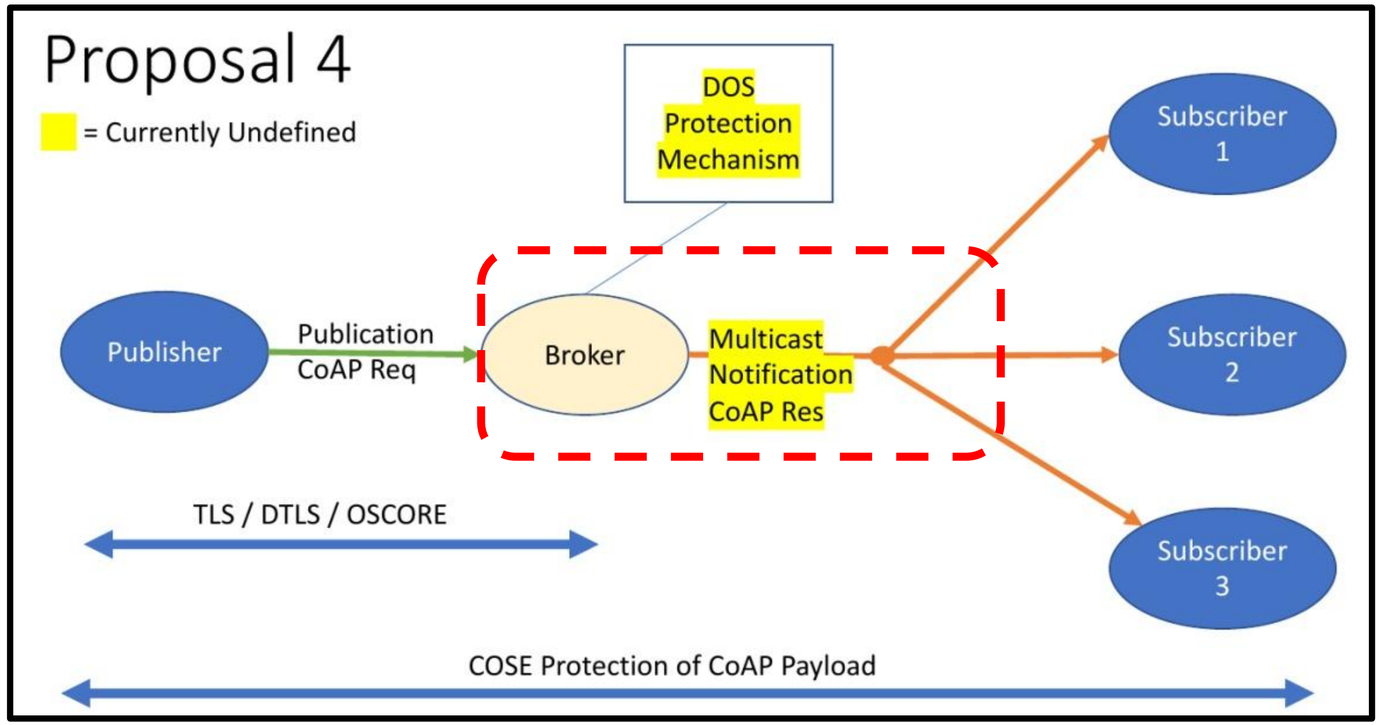
- Addition in *draft-dijk-core-groupcomm-bis*
  - One multicast registration request, to multiple Servers S1, S2, …
  - Separate notifications from S1, S2, …, as unicast responses

88

- Next step: notifications as multicast responses
  - Many clients observe the same resource on a Server S
  - Improved performance due to multicast delivery
  - Multicast responses are not defined. Token binding? Security?

Motivation (2/2)

- Practical use case
  - Pub-Sub scenario
  - Many clients subscribe to a same topic on the Broker
  - Use multicast for notifications



89

From the Hallway Discussion @ IETF 104

- Notifications as multicast responses?
  - Pros: better performance; subscribers can be client only
  - How? (again): multicast responses are not defined; token binding; security

Contribution

- Define Observe notifications as multicast responses

- Management and enforcement of a common Token space
  - Token space managed by the Server
  - Align all clients of an observation to use a same Token value

- Use of Group OSCORE to protect multicast notifications
  - The server aligns all clients of an observation on a same external_aad

Assumptions

› Clients have previously discovered resources to access

› Clients know the IP multicast address to listen to for notifications
  – E.g. through the CoRE Resource Directory and its Groups usage pattern

› If Group OSCORE is used to secure multicast notifications
  – Clients and Server have previously joined the right OSCORE group

Token overriding

- New CoAP Option “Override-Token”
  - Included by the Server in the first unicast notification to each client
  - Specifies a same Token value *T* used in multicast notifications for that resource

- Designed to work also through CoAP Proxies
  - Same approach on both client- and server- side of each Proxy
  - A proxy may remove the option and replace it with a new instance

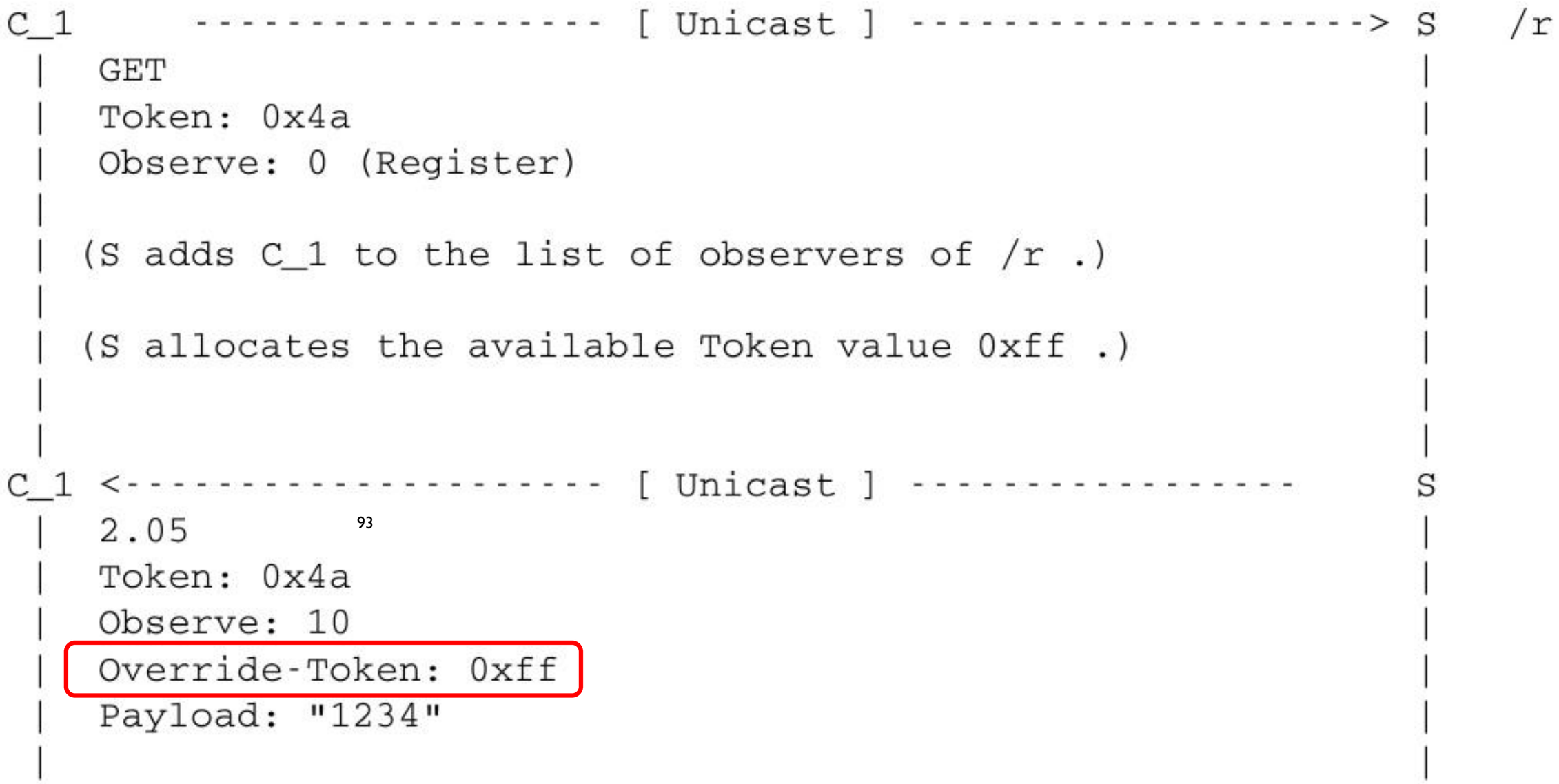
- Class U for OSCORE [92]

No.	C	U	N	R	Name	Format	Length	Default
TBD1	X	x	-		Override-Token	opaque	1-8	(none)

C=Critical, U=Unsafe, N=NoCacheKey, R=Repeatable

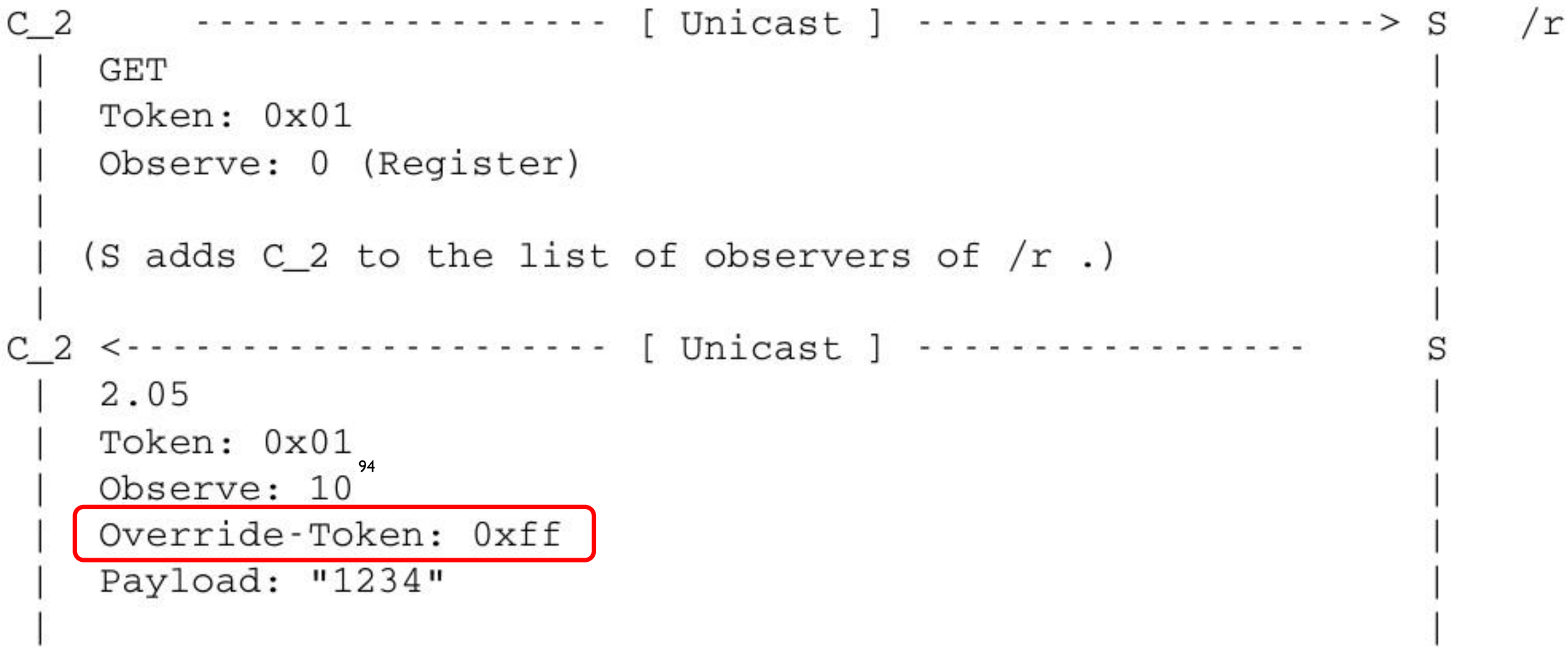
Figure 1: The Override-Token Option.

C1 registration

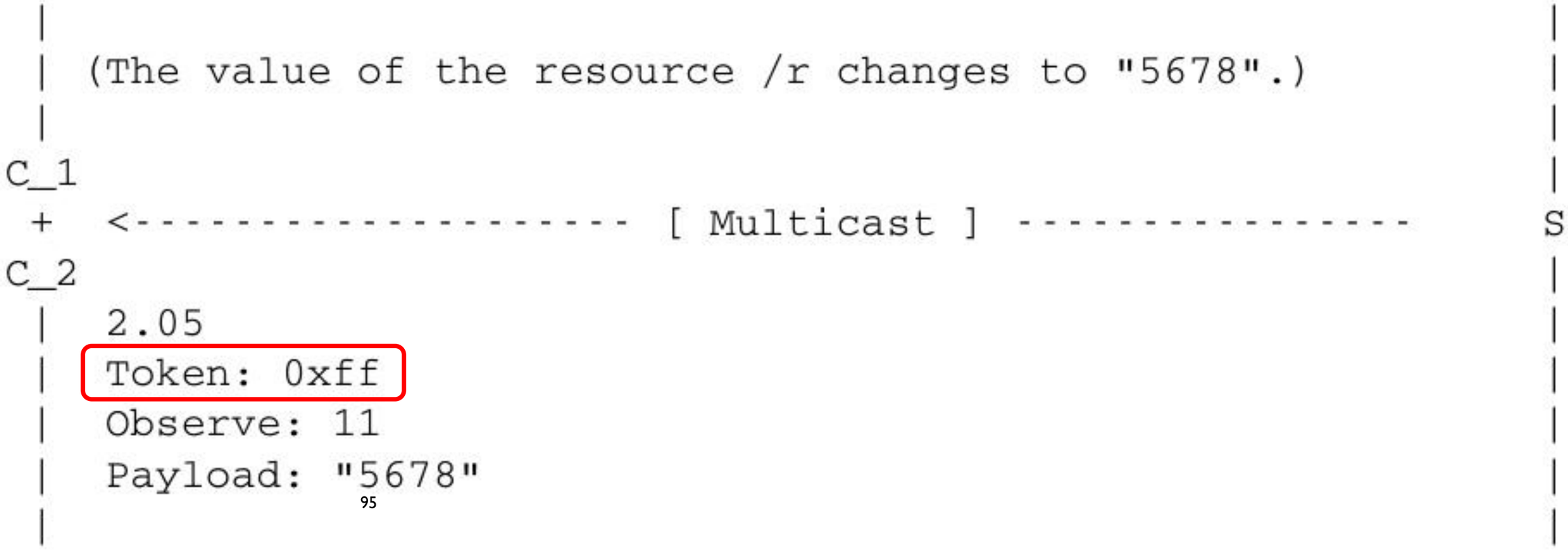


93

C2 registration



Multicast notification



95

- Enforce binding between
  - Every multicast notification for a resource
  - The corresponding observation of each client observing that resource

Range of Token values

- Server side
  - MUST use for multicast notifications
  - No reuse over multiple resources
- Client side
  - MUST NOT use for outgoing messages
  - Exception: cancellation of observation

Token size (Bytes)	Token value range	Range size
1	[0xf0 , 0xff]	16
2	[0xffc0 , 0xffff]	64
3	[0xffff00 , 0xffffff]	256
4	[0xffffbff , 0xffffffff]	1024
5	[0xfffffff7ff , 0xfffffffff]	2048
6	[0xffffffffefff , 0xfffffffffff]	4096
7	[0xffffffffffdfff , 0xffffffffffffff]	8192
8	[0xfffffffffffbfff , 0xffffffffffffffff]	16384

96

- Rationale
  - Distinction among a multicast notification and a regular (notification) response
  - Keep the clients able to disambiguate

Security with Group OSCORE

- New CoAP Option “Override-AAD”
  - Included by the Server in the first unicast notification to each client
  - Specifies a CBOR Array [*x* : bstr, *y* : int], the same to each client
  - *x* is the Sender ID (‘kid’) of the Server, in the OSCORE group
  - *y* is the SN of the Server, in the OSCORE group, when the first client registers
  - Note: the Server consumes the value *y* and does not reuse it as SN in the group

- To secure/verify all multicast notifications, the OSCORE *external_aad* is built with:
  - ‘req_kid’ = *x* and ‘req_piv’ = *y*

- Class E for OSCORE

97

No.	C	U	N	R	Name	Format	Length	Default
TBD2	X				Override-AAD	(*)	3-255	(none)

C=Critical, U=Unsafe, N=NoCacheKey, R=Repeatable
(*) See below.

Figure 2: The Override-AAD Option

Example with security

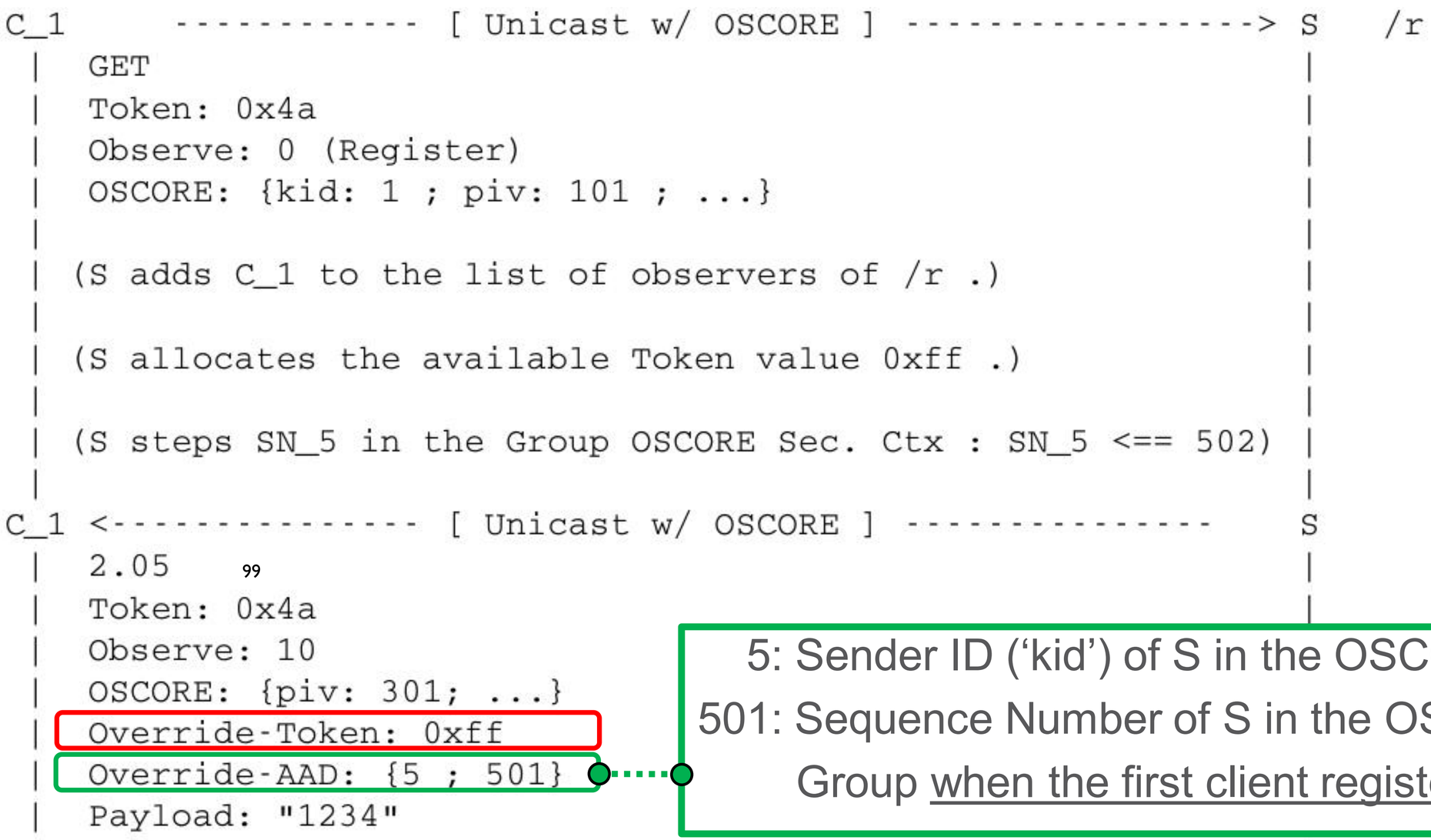
- {C_1, S} – OSCORE Initial status
  - C_1 : Sender ID ‘kid’ = 1; Sequence Number SN_1 = 101
  - S : Sender ID ‘kid’ = 3; Sequence Number SN_3 = 301

- {C_2, S} – OSCORE Initial status
  - C_2 : Sender ID ‘kid’ = 2; Sequence Number SN_2 = 201
  - S : Sender ID ‘kid’ = 4; Sequence Number SN_4 = 401

98

- {C_1, C_2, S} – Initial status in the OSCORE group
  - Group ID ‘kid_context’ = “feedca57ab2e”
  - S: Sender ID ‘kid’ = 5; Sequence Number SN_5 = 501

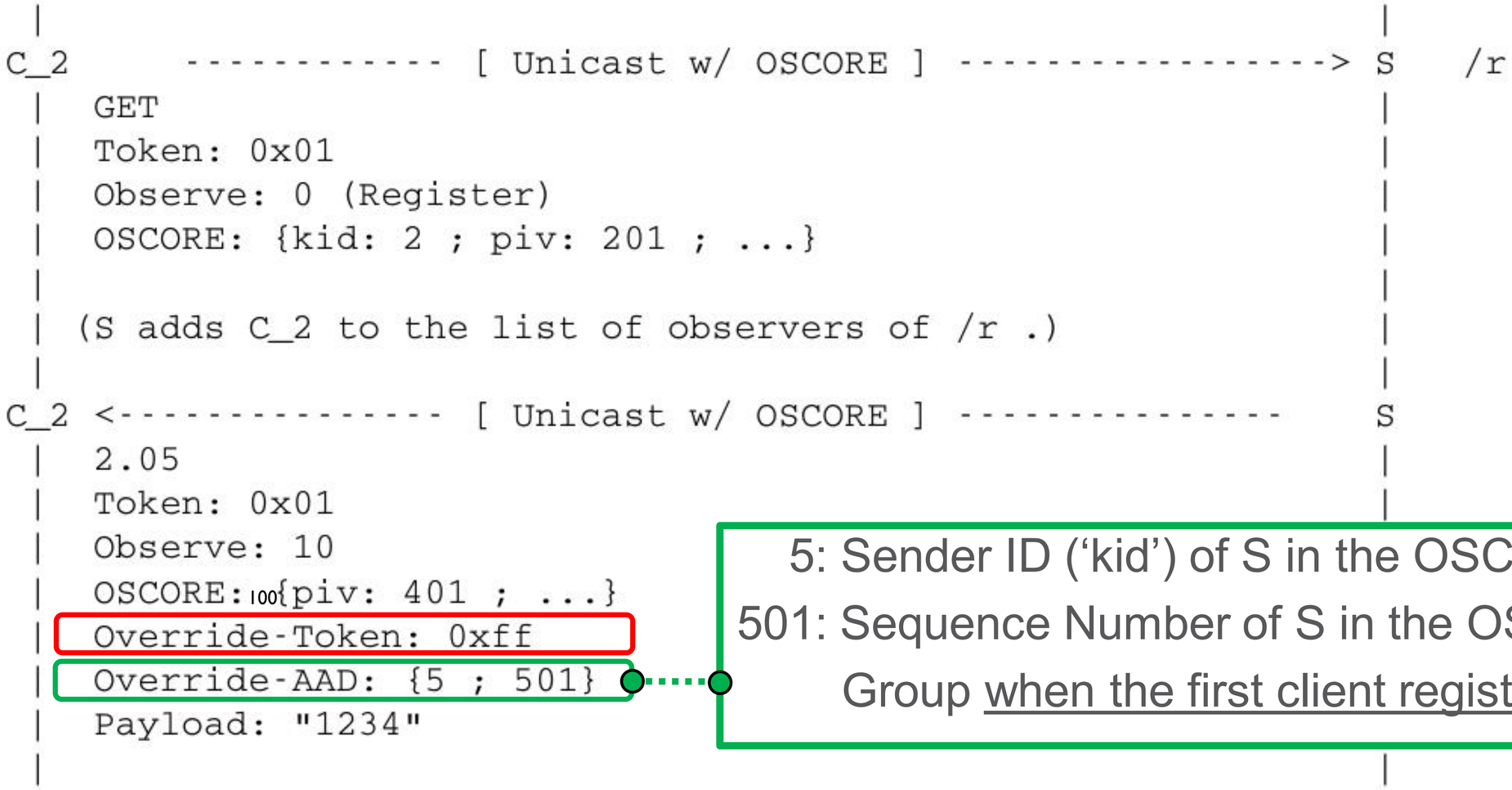
C1 registration w/ security



5: Sender ID ('kid') of S in the OSCORE Group

501: Sequence Number of S in the OSCORE Group when the first client registers

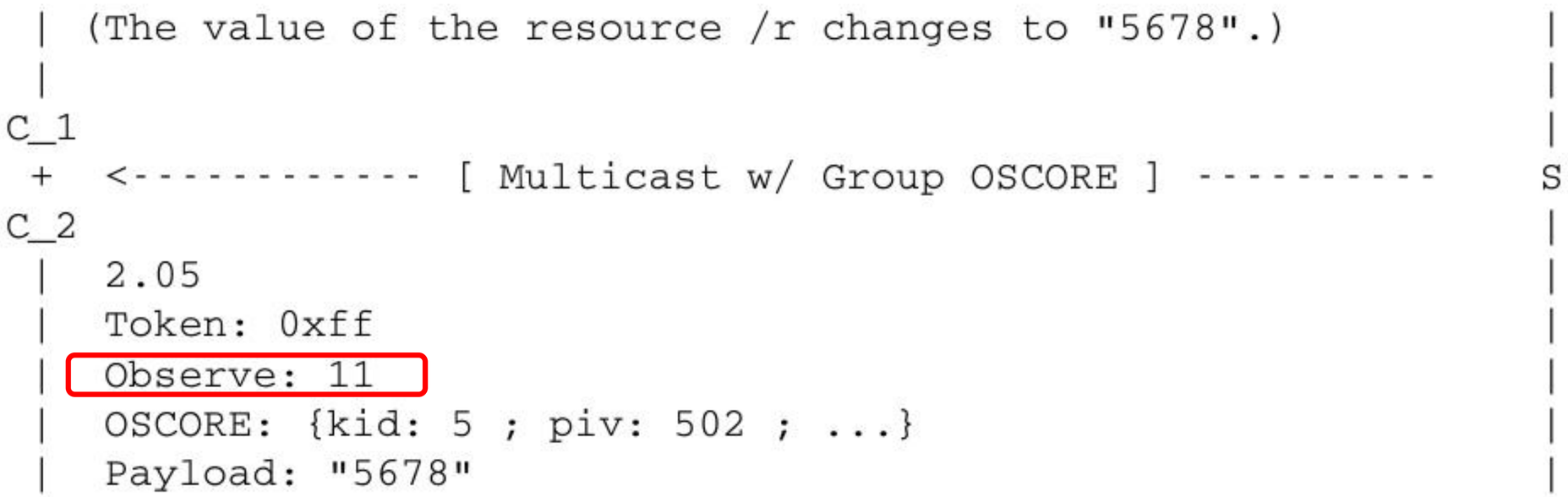
C2 registration w/ security



5: Sender ID ('kid') of S in the OSCORE Group

501: Sequence Number of S in the OSCORE Group when the first client registers

Multicast notification w/ security



- When encrypting and signing the multicast notification:
  - The OSCORE *external_aad* has 'req_kid' = 5 and 'req_iv' = 501
  - Same for all following notifications for the same resource
- Enforce secure binding between
  - Every multicast notification for a given resource
  - The first unicast notification sent to each client after registration

Summary

- Multicast notifications to all clients observing a resource
  - The server synchronizes the Token value upon Clients' registration
  - All notifications can be securely bound to the registration response

- Benefits
  - Better performance when many clients observe a same resource
  - In pub-sub scenarios, subscribers can be only clients

- Need for document reviews

Thank you!

Comments/questions?

https://gitlab.com/crimson84/draft-tiloca-core-observe-responses-multicast

Backup

Contribution

- Define Observe notifications as multicast responses
  1. Management and enforcement of a common Token space
  2. Use of Group OSCORE to protect multicast notifications

- Token space managed by the Server
  - A same Token value *T* is provided to all observers of a same resource
  - *T* is provided in the registration response to each client, over unicast
  - Multicast notifications for that resource use *T* as their Token value

105

- When using Group OSCORE
  - The Server provides same parameters to the observers of a same resource
  - Provisioning of parameters in the registration response to each client, over unicast
  - Those parameters are used when protecting multicast notifications for that resource

All times are in time-warped EDT (UTC–04:00)

Thursday (120 min)

- 10:00–10:05 Intro, Agenda
- 09:05–09:20 OSCORE groupcomm (MT)
- 09:20–09:30 OSCORE discovery (MT)
- 09:30–09:45 Observe multicast notifications (MT)
- 09:45–10:00 Groupcomm bis (ED)
- 10:00–10:10 SenML data ct (AK)
- 10:10–10:20 SenML local base name (HT)
- 10:20–10:30 SenML units (CB)
- 10:30–10:35 Fasor
- 10:35–11:00 Flextime

Group Communication for the Constrained Application Protocol (CoAP)

draft-dijk-core-groupcomm-bis-01

Esko Dijk, IoTconsultancy.nl
Chonggang Wang, InterDigital
Marco Tiloca, RISE

IETF 105, CoRE WG, Montreal, July 25th 2019

Goal (recap from IETF104)

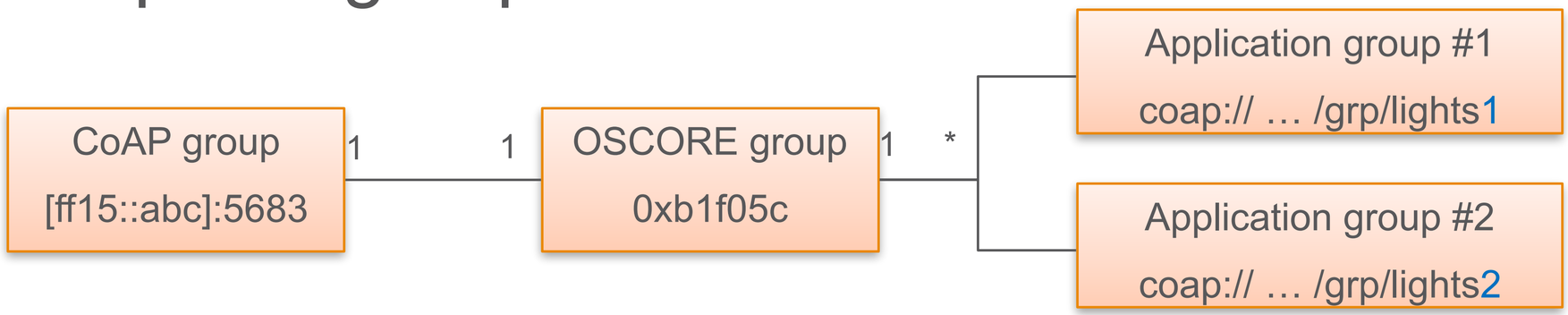
- Intended normative successor of experimental RFC 7390 (if approved)
  - As a Standards Track document
  - Refer to RFC 7390 when possible
- Be standard reference for implementations that are now based on RFC 7390, e.g.:
  - “Eclipse Californium 2.0.x” (Eclipse Foundation)
  - “Implementation of CoAP Server & Client in Go” (OCF)
- What’s in scope?
  - CoAP group communication over UDP/IP, including latest developments (Observe/Blockwise/ …)
  - Both unsecured and group-OSCORE-secured
  - Principles for secure group configuration
  - Use cases

Groupcomm-bis-01 (1/3)

- Updated using part of review comments
  - More updates based on reviews pending (thanks reviewers!)
- Scope clarified
- Decision to “copy over” RFC 7390 content
  - Updates to content applied where needed
- OSCORE group key management detailed (5.2.1)
- Use cases moved to Appendix (informative part)
- Bugfixes

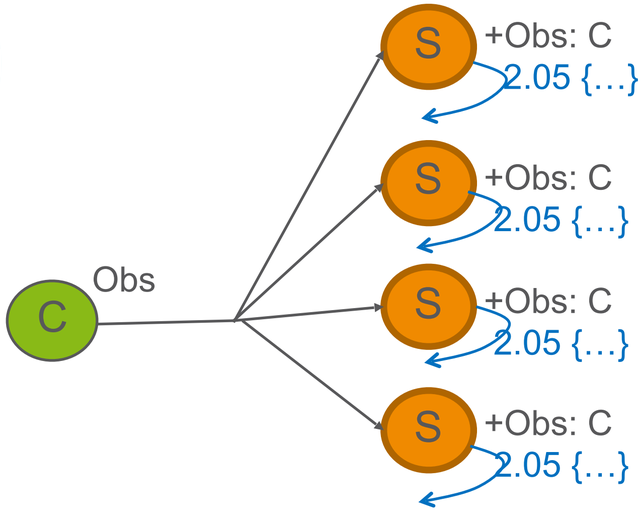
Groupcomm-bis-01 (2/3)

- Distinguish *types* of groups (identifiers for group type:)
  - CoAP group: network level — multicast-address + port
  - OSCORE group (‘security group’) — Group ID (Gid) + Master Secret
  - Application group: application level — <any application-specific ID>
- *To do: relations between group types can be detailed*
- Example of group relations:



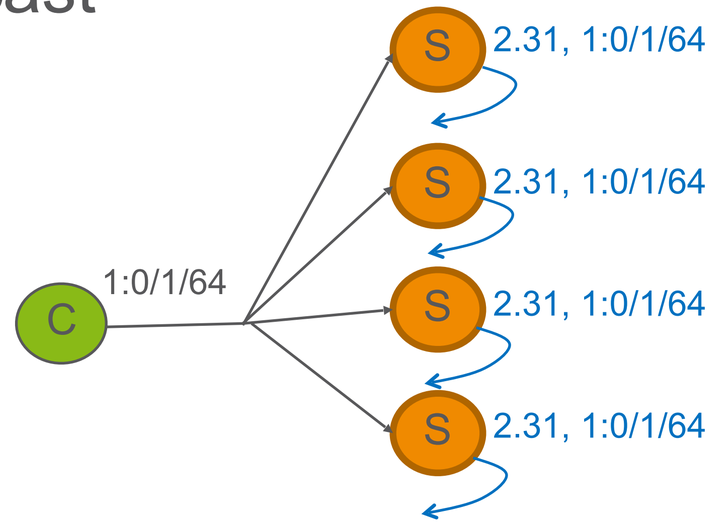
Groupcomm-bis-01 (3/3)

- Detailed multicast GET + Observe
  - Use case: observe a resource in a group



- Detailed multicast PUT / POST / (i)PATCH + Block1
  - Use case: efficient software distribution over multicast
  - While writing, was found to be non-trivial
  - **Decide**: to “spin out” to new I-D ?

- Above updates the RFCs 7641 & 7959 – see slide 7



New scope & approach (1/2)

- Obsoletes: RFC 7390
  - Changed from '*updates*' to '*obsoletes*' in -01
  - Standards track I-D that *updates* experimental RFC would lead to unclarity what is normative vs experimental
  - Should 'copy over' any 7390 content we wish to keep
  - The experimental REST group configuration interface remains experimental in 7390 (that part not obsoleted!)

New scope & approach (2/2)

- Updates: RFC 7641 – Observe
  - With: Multicast GET + Observe use
  - RFC 7641 does not define multicast use
- Updates: RFC 7959 – Block-wise transfer
  - With: Multicast PUT / POST / (i)PATCH + Block1
  - RFC 7959 only considers multicast GET + Block2
- Updates: RFC 7252 – CoAP
  - With: new CoAP client request/response matching rule
  - This will likely be removed again – based on a misinterpretation of ‘client’

Next steps

- Complete processing of review comments
- Implement scope decisions – multicast Block1 spin-out?
- Complete the document
  - Copy over & update remaining relevant sections of RFC 7390
  - Replace TBDs with actual content
  - Add possibly missing points. Any input?
- Need for document reviews on upcoming -02 update

Thank you!

Comments/questions?

https://gitlab.com/crimson84/draft-groupcomm-bis

Motivation (backup slide)

- RFC 7390 was published in 2014
  - CoAP functionalities available by then were covered
  - No group security solution was available to indicate
  - It is an Experimental document (started as Informational)

- What has changed?
  - More CoAP functionalities have been developed (Block-Wise, Observe)
  - RESTful interface for membership configuration is not really used
  - Group OSCORE provides group end-to-end security for CoAP

- Practical considerations
  - Group OSCORE clearly builds on RFC 7390 normatively
  - However, it can refer RFC 7390 only informationally

All times are in time-warped EDT (UTC–04:00)

Thursday (120 min)

- 10:00–10:05 Intro, Agenda
- 09:05–09:20 OSCORE groupcomm (MT)
- 09:20–09:30 OSCORE discovery (MT)
- 09:30–09:45 Observe multicast notifications (MT)
- 09:45–10:00 Groupcomm bis (ED)
- 10:00–10:10 SenML data ct (AK)
- 10:10–10:20 SenML local base name (HT)
- 10:20–10:30 SenML units (CB)
- 10:30–10:35 Fasor
- 10:35–11:00 Flextime

Fast-Slow Retransmission Timeout and Congestion Control Algorithm for CoAP

draft-jarvinen-core-fasor

Ilpo Järvinen, **Markku Kojo**, Iivo Raitahila, Zhen Cao

IETF105

July 25, 2019

FASOR

- FASOR is an alternative Retransmission Timeout (RTO) and congestion control algorithm for CoAP
- Optional to implement in CoAP (like CoCoA)
- Implemented for libcoap
- Experimentally evaluated & results published [1]

[1] I. Järvinen, I. Raitahila, Z. Cao, M. Kojo, "FASOR Retransmission Timeout and Congestion Control Mechanism for CoAP", In Proc IEEE Global Communications Conference (Globecom 2018), Abu Dhabi, UAE, December 2018.

I-D History

- -00 published in July, 2018
  - presented in ICCRG@IETF-102
  - not enough session time in CoRE@IETF-102 to present
- -01 published in Oct 2018
  - added pseudocode + some small clarifications
  - presented in CoRE@IETF-103
- -02 published in July 2019,
  - no modifications, just refreshed the doc

FASOR Status

- Intended status: Experimental (→ PS)
  - RFC 5033 (BCP 133) - Specifying New Congestion Control Algorithms:

    ... alternate congestion control algorithms are expected to be published as "Experimental“ RFCs until such time that the community better understands the solution space.

- Ready for WG adoption?

draft-jarvinen-core-fasor “IPR”

IETF104 slide

- For draft-jarvinen-core-fasor, patent claims were laid out in https://datatracker.ietf.org/ipr/3227/
- At IETF103, we said that the information given might not be sufficient to make a WG decision on its impact
- The statement has since been updated: https://datatracker.ietf.org/ipr/3346/
  - Not asserted for “essential” part of IETF *standard*
  - But under reciprocity (“defensive patent”)
  - (FRAND available, too)
- Do we now have sufficient information, to discuss, e.g., working group adoption?

All times are in time-warped EDT (UTC–04:00)

Thursday (120 min)

- 10:00–10:05 Intro, Agenda
- 09:05–09:20 OSCORE groupcomm (MT)
- 09:20–09:30 OSCORE discovery (MT)
- 09:30–09:45 Observe multicast notifications (MT)
- 09:45–10:00 Groupcomm bis (ED)
- 10:00–10:10 SenML data ct (AK)
- 10:10–10:20 SenML local base name (HT)
- 10:20–10:30 SenML units (CB)
- 10:30–10:35 Fasor
- 10:35–11:00 Flextime

SenML Data Value Content-Format Indication

draft-keranen-core-senml-data-ct-02

Ari Keränen

IETF 105

Content-Format indication

- SenML Records can contain (binary) "data values" in a "vd" field

```
[
 {"bn":"urn:dev:ow:10e2073a01080063:", "n":"temp", "v":7.1},
  {"n":"open", "vb":false},
  {"n":"nfc-reader", "vd":"aGkgCg"}
]
```

- This draft: new Content-Format indication ("ct") field to indicate the Content-Format of the data in the SenML Record

Example SenML Record with data value and Content-Format indication

```
{"n":"nfc-reader", "vd":"gmNmb28YKg", "ct":60}
```

Example SenML Record with data value and Content-Format indication

```
{"n":"nfc-reader", "vd":"gmNmb28YKg", "ct":60}
```

CBOR CoAP Content Format

base64(

```
82          # array(2)
  63        # text(3)
    666F6F  # "foo"
  18 2A     # unsigned(42)
```

)

Content-Type and Content-Coding

- Not all Media-Types and Content-Coding alternatives (will) have CoAP Content-Format IDs assigned
  - Need content-type and content-coding capabilities too
  - **But** new fields with base fields makes processing complicated (see IETF 104)
- **Latest** proposal: "ct" for both numeric and string format
  - Using "@suffix" for content-coding (if any)

```
{"n":"nfc-reader-42",
 "vd":"H4sIAA+dmFwAAzMx0jEZMAQALnH8Yn0AAAA",
 "ct":"text/csv@gzip"}
```

Numbers as strings (actually makes sense)?

- All SenML fields (so far) have single, fixed type
  - e.g., "v" for numeric values and "vs" for string values
  - More efficient to parse in high speed (no type checks needed)
  - Easier to automatically generate parser code
  - Easier to store in database with a simple schema
- "ct" values can be numeric (CoAP ct IDs) or strings (content-types)
  - Potential solution: express both as strings: `"ct":"60"`
  - Only ~2 bytes more overhead (quotes in JSON, string vs. uint in CBOR)
  - Usually equality-comparison and no arithmetic needed anyway

Questions

- Any concerns on using string value for content-formats?

- Comments/concerns on combined content-type and –coding syntax?
  - "application/json@deflate"
  - potential bike shed: delimiter character

- Minor issue: add must-understand-ct-field (i.e., "ct_")?
  - could send alternating content-formats from the same sensor
  - and there's no really additional cost specifying that

- Ready for WGLC?

Extra question: mixing b and _ fields: what are the resolution rules?

1)

```
[
 {"bfoo_":42, "n":"t1", "v":1},
 {            "n":"t2", "v":2}
 {"foo": 1,   "n":"t3", "v":3}
]
```

2)

```
[
 {"bfoo_":42, "n":"t1", "v":1},
 {            "n":"t2", "v":2}
 {"foo_": 1,  "n":"t3", "v":3}
]
```

3)

```
[
 {"bfoo":42, "n":"t1", "v":1},
 {           "n":"t2", "v":2}
 {"foo_": 1, "n":"t3", "v":3}
]
```

4)

```
[
 {"bfoo":42, "n":"t1", "v":1},
 {           "n":"t2", "v":2}
 {"foo": 1,  "n":"t3", "v":3}
]
```

All times are in time-warped EDT (UTC–04:00)

Thursday (120 min)

- 10:00–10:05 Intro, Agenda
- 09:05–09:20 OSCORE groupcomm (MT)
- 09:20–09:30 OSCORE discovery (MT)
- 09:30–09:45 Observe multicast notifications (MT)
- 09:45–10:00 Groupcomm bis (ED)
- 10:00–10:10 SenML data ct (AK)
- 10:10–10:20 SenML local base name (HT)
- 10:20–10:30 SenML units (CB)
- 10:30–10:35 Fasor
- 10:35–11:00 Flextime

Local Basename

Hannes Tschofenig

The Basename

Current

(globally unique identifier in each SenML payload)

```
[
  {"bn":"urn:dev:ow:10e2073a01080063:",
    "n":"temp","u":"Cel","v":23.1},
  {"n":"label","vs":"Machine Room"},
  {"n":"open","vb":false},
  {"n":"nfc-reader","vd":"aGkgCg"}
]
```

New

(local basename)

```
[
   {"lbn":"/3303/", "n":"0/5700/", "v":25.2},
   {"n":"/1/5700/", "v":5}
]
```

All times are in time-warped EDT (UTC–04:00)

Thursday (120 min)

- 10:00–10:05 Intro, Agenda
- 09:05–09:20 OSCORE groupcomm (MT)
- 09:20–09:30 OSCORE discovery (MT)
- 09:30–09:45 Observe multicast notifications (MT)
- 09:45–10:00 Groupcomm bis (ED)
- 10:00–10:10 SenML data ct (AK)
- 10:10–10:20 SenML local base name (HT)
- 10:20–10:30 SenML units (CB)
- 10:30–10:35 Fasor
- 10:35–11:00 Flextime

Additional Units for SenML

(Slides donated by Ari, thank you)

SenML Units Registry today

- Registry of short strings to represent measurement units
  - e.g., "m" for meters and "s" for seconds
- Restrictive policy to facilitate interoperability
  - Unscaled SI units and "a few more"
  - Only one unit for each kind of measurement (no "km" or "miles")
- However, many derived and other units in practical use today
  - OMA SpecWorks IPSO/LwM2M models use a richer set:
    "ms" for time values, also "kWh", "dBm", etc.
  - Would need to change a large amount of existing models to use unscaled SI
  - Many use cases have a "natural" scaled/offset unit (e.g., "ms" for time or "um" for particle size); having to use exponent every time brings extra cost

Proposal: additional registry

- Another SenML IANA (sub) registry for units with more relaxed rules
  - recommending to use the restricted set for best interoperability
  - Relaxed registry would describe translation rules for conversions to the restricted set, for example:

"Relaxed unit"	SenML unit	scale	offset
km	m	1000	0
dBm	dBW	1	-30

- Discussed at the joint meeting with OMA SpecWorks with in-room consensus that this was a good way forward

Flextime

Small calls for participation

to pre- or non-draft activities around CoAP

Christian Amsüss

2019-07-25

SCHC for inside CoAP

Motivation	Compression or avoiding (eg. decimal) parsing
Example	*IPSO-Path* binary `cee4` `0000` `1644` instead of *Uri-Path* as ASCII `3300` / `0` / `5700` "
Came up	in LwM2M, A-REaLiST, complaints about long `.well-known/protocol-name/...` names
Experience	is in SCHC, use that
Draft	None yet, won't start alone
Details	open (High-number options? High-Content-Format Payload?)

Programming-language level APIs to CoAP messages

Scope	manipulating a single CoAP message
Motivation	developing portable CoAP tools
Methods	“set code”, “iterate over options”
Platforms	language and scale independent
Audience	CoAP implementors, not standardization

CoAP FAQ

- Wiki page with non-authoritative answer to repeated questions
- Answers can point to official (LWIG? corr-clar?) answers later
- `https://github.com/core-wg/wiki/wiki/CoAP-FAQ`