Network Working Group
Internet-Draft
Intended status: Informational
Expires: January 5, 2015

Y. Hong
Y. Choi
ETRI
D. Kim
M. Khan
W. JIN
Jeju Nat. Univ.

July 4, 2014

CoAP Endpoint Unit Identification for Multiple Sensor and Actuator in a Node

draft-hong-core-coap-endpoint-unit-id-00

Abstract

The Constrained Application Protocol (CoAP) is a protocol intended towards devices which are constrained in terms of memory, processing and power i.e. small low power sensors, switches and valves etc. The CoAP allows such devices to interactively communicate over the Internet. This document is motivated by the concept of a composite CoAP node, a single CoAP entity which integrates multiple CoAP resources (sensors, actuators) and the scheme to allow the identification of individual integrated resources while using the Unit ID as a new CoAP option. The Unit ID option in the CoAP enables the usage of composite nodes consisting of multiple sensors and actuators while having a single IP address for communication. The integrated resources can be individually or collectively communicated with and/or controlled using CoAP messages with additional options of UnitSize and UnitID. The UnitSize is basically a numeric value indicating the number of sub-resources in a composite CoAP node while the UnitID option has the string identifiers for the sub-resource(s) for which the message is intended. These options will enable the CoAP to communicate and control multiple resources by using single composite messages i.e. UnitID = "*", efficiently utilize IP addresses i.e. one IP multiple IDs, reduce communication traffic and hence conserve power among the CoAP resources.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Copyright Notice

Table of Contents

1. Introduction

This draft presents a conceptual architecture and design features of multiple Unit IDs in a node for resource discovery, registration and lookup. The concept of node ID has been presented in [I-D.li-coap-nodeid]. This draft presents the idea of nodes having

multiple integrated sensing and/or actuating devices. Each of these devices is separately identifiable via a Unit ID. The Unit ID for a given resource must be unique among all the integrated resources in a single node while the same ID can represent a resource integrated in another node.

The integrated resources inside a node are separately identified by node ID and Unit ID together. Every node has an IP address through which it can communicate with clients or other modules of the system (Resource Directory). A detailed description of the purpose and features of Resource Directory have been presented in [I-D.ietf-core-rd].

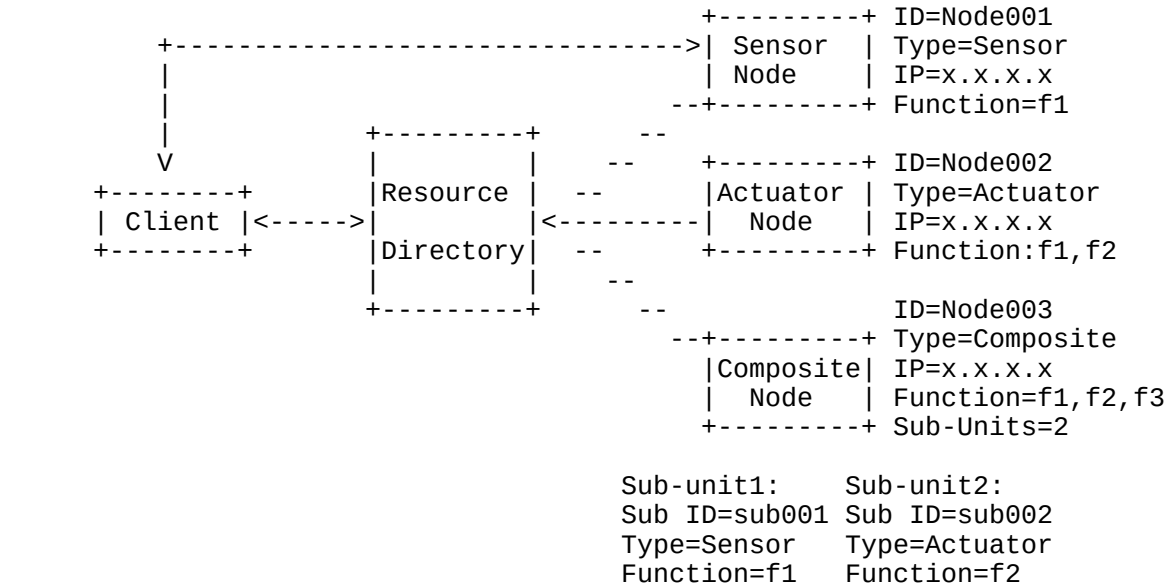

Figure 1: Endpoint Unit ID and Resource Directory

Figure 1 shows that a node may contain a single or multiple integrated resources i.e. multiple sensors, multiple actuators or sensors and actuators in a single node. The nodes register these resources with the Resource Directory. The Resource Directory defines its own function sets for discovery, registration and lookup etc. Once a node had registered all its integrated resources with the Resource Directory, the clients may lookup single or multiple resources and may interact with them directly. The Resource Directory helps in the automated discovery and lookup of resources

while the multi-Unit IDs provide an efficient utilization of a single IP for interacting with multiple resources.

As described in [RFC7252], there are two entities required for CoAP communication i.e. CoAP Client and CoAP Server. A CoAP Server may also act as client and vice versa if both of these entities have resources to share and require certain resources from each other. The CoAP server discovers a Resource Directory (RD) [I-D.ietf-core-rd]. The discovery of RD means finding location of the register function set in the RD using which a CoAP server may register the resources which it wants to share.

Once a complete path is obtained for a register function set in the RD, the CoAP server may then register (publish) resources to the RD. The CoAP clients then requests the RD to look up for registered resources. The RD then returns the access paths for the registered resources according to the request of the client. The returned resources may include simple or composite resources and the client can communicate with these resources. If a single CoAP node has multiple integrated sub devices, then the composite interaction with the resources is based on UnitID(s). The client can interact with individual sub devices or collectively interact with all the sub devices of a composite node. It is important to note that the description and discovery of resources hosted by a constrained web server is specified by the CoRE Link Format [RFC6690] which is based on the Web Linking [RFC5988] for the discovery of resources hosted by an HTTP Web Server.

2. Conventions and Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

3. The use case of multiple CoAP unit identification and control

Figure 2 shows the use case scenario for a CoAP composite node which integrates a light sensor and two switches to control the lights in a room. The composite node is accessed via a single IP address assigned to it while the sub-resources of the composite node are accessed with Unit IDs. The composite node like a normal CoAP Endpoint, registers its resources in the form of sub units with the RD. The RD, thus have a single IP address for the composite node and Unit IDs for the sub units of the composite node.

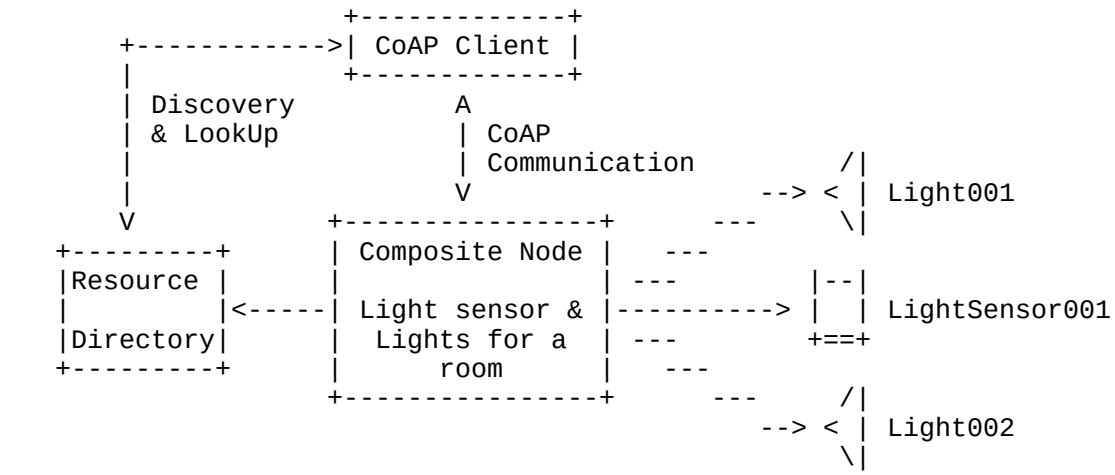

Figure 2: Multiple UnitID based composite CoAP node interaction use case

A CoAP client performs look up on the RD and gets the required resource information. Suppose the client wants to interact with the composite node, the information regarding all its sub units is also provided to the client by the RD. The client then use this information (i.e. UnitSize, UnitID) to create CoAP messages in order to interact with a single or multiple sub units of the composite node. For example, the user may send a CoAP message with UnitSize=1 and UnitID= "lightSensor001" to request data from the light sensor. The Composite node will return an ACK message with UnitID parameter and sensor reading as message payload. The client may also send a CoAP message with UnitSize=2 and UnitID= "Light001", UnitID="Light002" options to turn-on or off the lights with a single message.

4. IP and Endpoint Unit ID mapping architecture

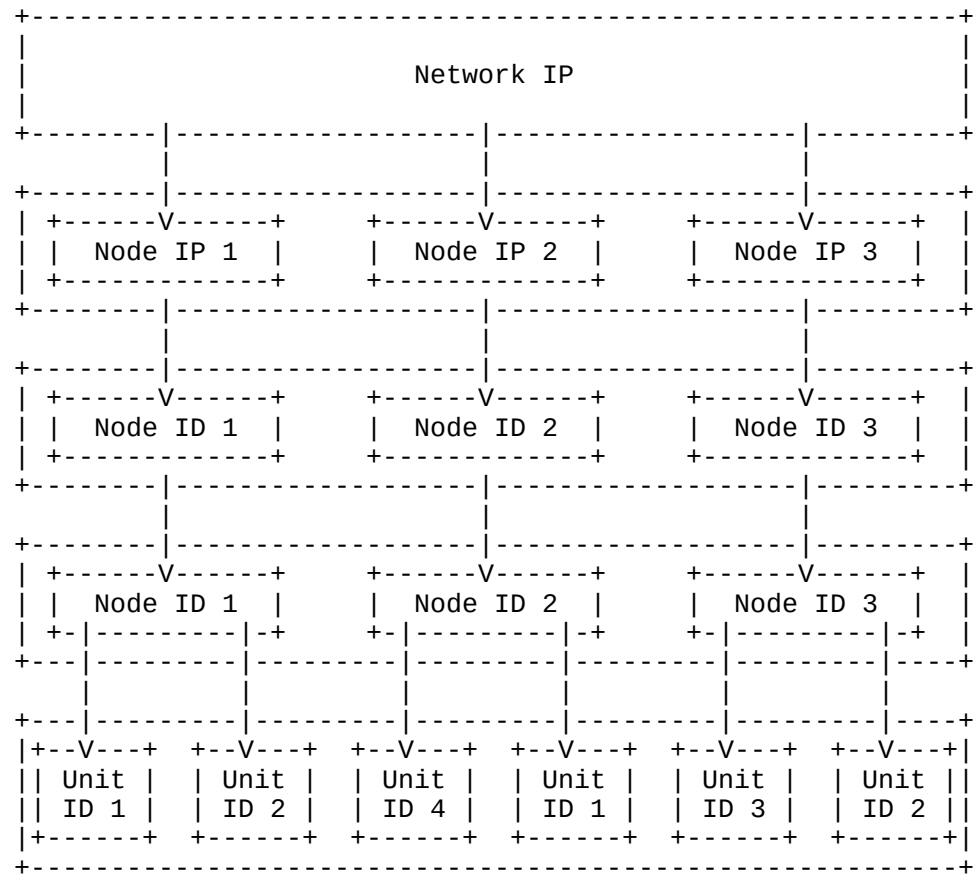

Figure 3: IP address and Endpoint Unit ID mapping architecture

Figure 3 presents a generalized architecture for IP and ID mapping in the proposed Endpoint Unit ID scenario. The network IP and local IP addresses are used to access the network of the node and the physical node respectively. In the CoAP a node ID is used to insure the consistency of the communication when an IP address change at the client or the server occurs during a communication session. Thus a node IP address and node ID pair used to communicate with a single resource. We propose that a single node may have multiple integrated resources and each of these resources can be represented by multiple sub-identifiers (IDs). The sub-identifier for the integrated resource is called as the Unit ID and a node may have more than one Unit IDs.

This scheme enables the use of single IP address for communicating with multiple resources (units) and each resource may be treated as a separate entity having its own address. Thus the result is efficient utilization of addressing space by combining the Node IP and Unit ID pairs. Group registration, lookup etc. and group communication for CoAP resources have been described in [I-D.ietf-core-rd] and [I-D.ietf-coap-group] respectively but both these drafts consider every resource in a group as a unique addressable entity hence no benefits when it comes to controlling IP address space usage or communication traffic load.

5. Benefits of the Endpoint Unit Identification

The Unit ID concept for composite endpoint (Node) provides the following major benefits.

a. A composite node with multiple integrated sub-unit resources will require only one IP address and using the IP address and Unit ID pairs, individual resources can be separately accessed without the need to have a separate IP address for each resource. Thus the proposed scheme efficiently utilizes IP address space to represent more devices with lesser number of IP addresses.

b. A single CoAP message with Unit ID parameter may be used to control sub-devices collectively using special characters. For example, a given composite endpoint may have sensors and actuators and all these sub-unit devices can be controlled with a single message using "*" as the Unit ID parameter value.

c. Using composite messages for Unit ID may also benefit in reducing traffic flow between client and endpoints (CoAP Server) and may also help in conserving energy in the constrained devices.

6. The Extended CoAP Header

Figure 4 shows the CoAP message header format. The header for the CoAP message is all the same with fields such as version, Type, and Token length etc. The change can be seen in the options section where the UnitSize field specifies the number of sub-unit integrated into a single composite node and the UnitID option which can hold a string ID for UnitID representing a sub-unit in a composite node. The UnitID field can be repeated multiple times according to the value of the UnitSize parameter and every time representing a single string ID for a sub-unit related to a specific composite node.

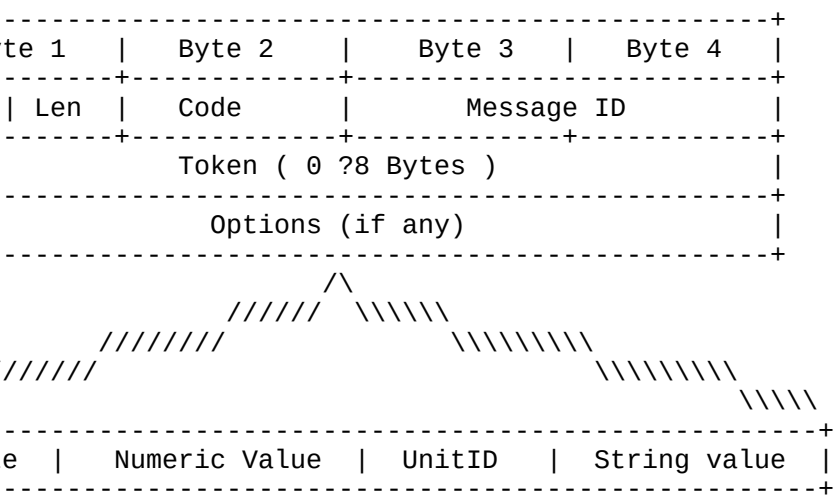

Figure 4: Multi-ID CoAP message format

7. Procedure of the Endpoint Unit Identification

7.1. Sub Unit(s) Registration with RD

Figure 5 shows the sequence of activities involved in the registration of Endpoint Unit ID nodes with integrated resources in the RD. In order for a node to register its integrated resources with the RD, the node uses the RD's registration function set and sends a CoAP POST message to the RD. The message payload contains the list of all the Unit IDs associated with the node. The RD receives the message and checks whether the request is valid. If the RD receives a valid request from the node, the source IP address and Port number from the CoAP request parameters or the message source address portion (default). The RD then extracts the Unit IDs from the message payload and creates a resource location for all the resources and returns a response message to the node. If the registration process is successful then a location URI is returned to the requesting node so it may update the registration or remove the location entry thus cancelling the registration of its integrated resources otherwise an error message is returned mentioning the cause of the failure.

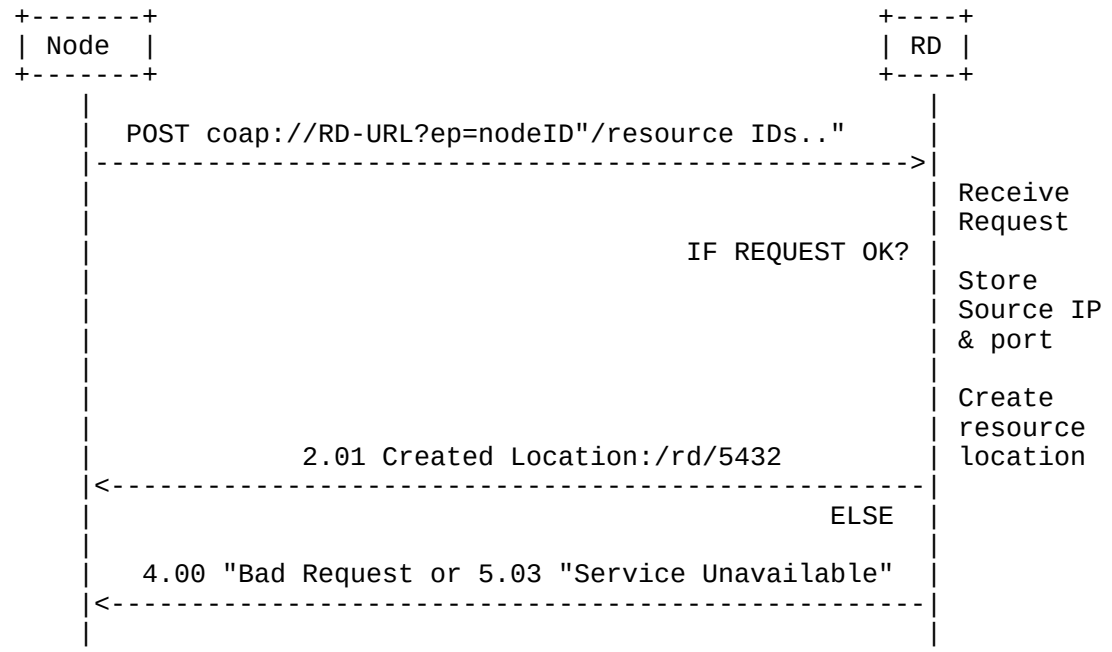

Figure 5: Endpoint Unit ID resource registration with RD

7.2. Sub Unit Lookup in RD

Figure 6 presents the RD based lookup process for Endpoint Unit ID resources integrated into a single node i.e. single IP address. The diagram shows a client requesting for a specific type (Temperature) of resources registered with the RD. For this purpose, it sends GET request to the RD with the type of resources the client wants to lookup in the directory. The RD receives the message, checks if the message is a valid CoAP request and then gets the IDs for all the registered resources with the resource type value equivalent to the one requested by the client (Temperature). The RD then creates a response message with the list of node IP address and resource IDs and sends it to the client. The client may then choose a specific resource from this list and communicate with it directly using the CoAP protocol.

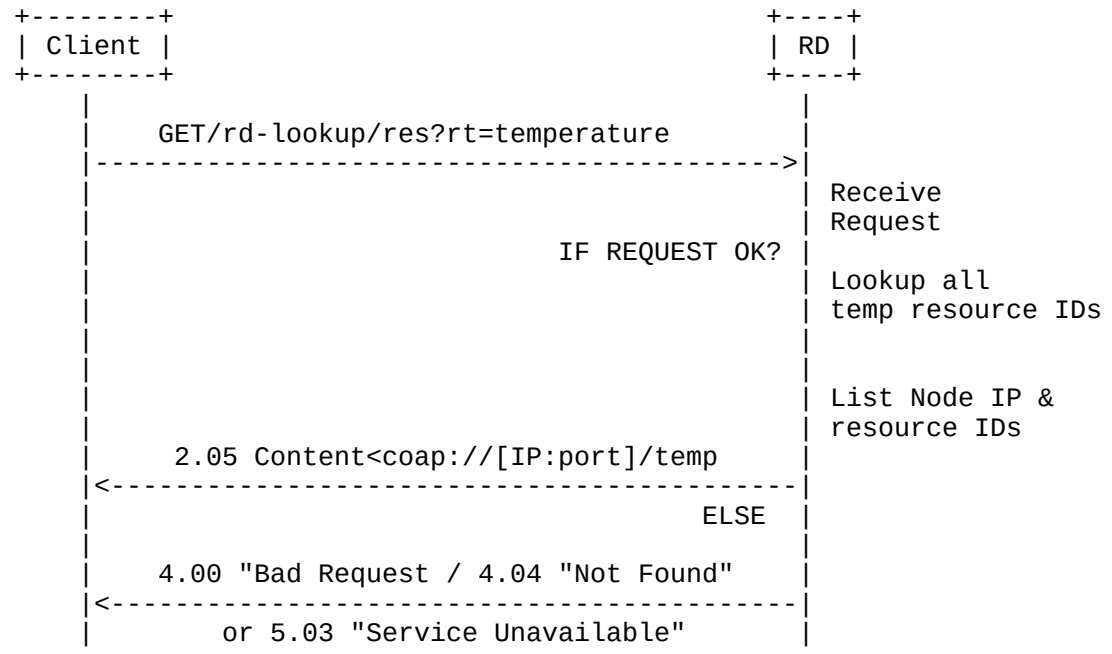

Figure 6: RD based resource lookup

7.3. CoAP Client Server Interaction (Single Unit)

Figure 7 shows the interaction among a client and resource (CoAP Server). As mentioned previously, the client performs lookup on the RD for a specific resource type and gets the list of all the resource IDs (node ID and Unit ID) registered with the RD. The following figure shows the process of client selecting a resource from that list and communicating with it directly.

Once the client decides to interact with a resource, it gets the resource complete URI i.e. Node IP address, Port number and Unit ID if it is a composite node. For a simple resource i.e. sensor or actuator, the node ID is used in conjunction with the IP address to perform the interaction between the CoAP client and server while for a composite node i.e. with multiple integrated resources (multiple IDs), the client creates a Unit ID, Token pair and sends a GET request to the integrated resource of a node using the complete URI. Here the Token means the CoAP token sent with a normal GET request. The node (CoAP Server), checks the request’s validity and responds back to the client with an ACK, consisting of the Token and data from

the integrated resource. The client checks the source of the data by comparing the Token of the ACK with the stored Unit ID, Token pair.

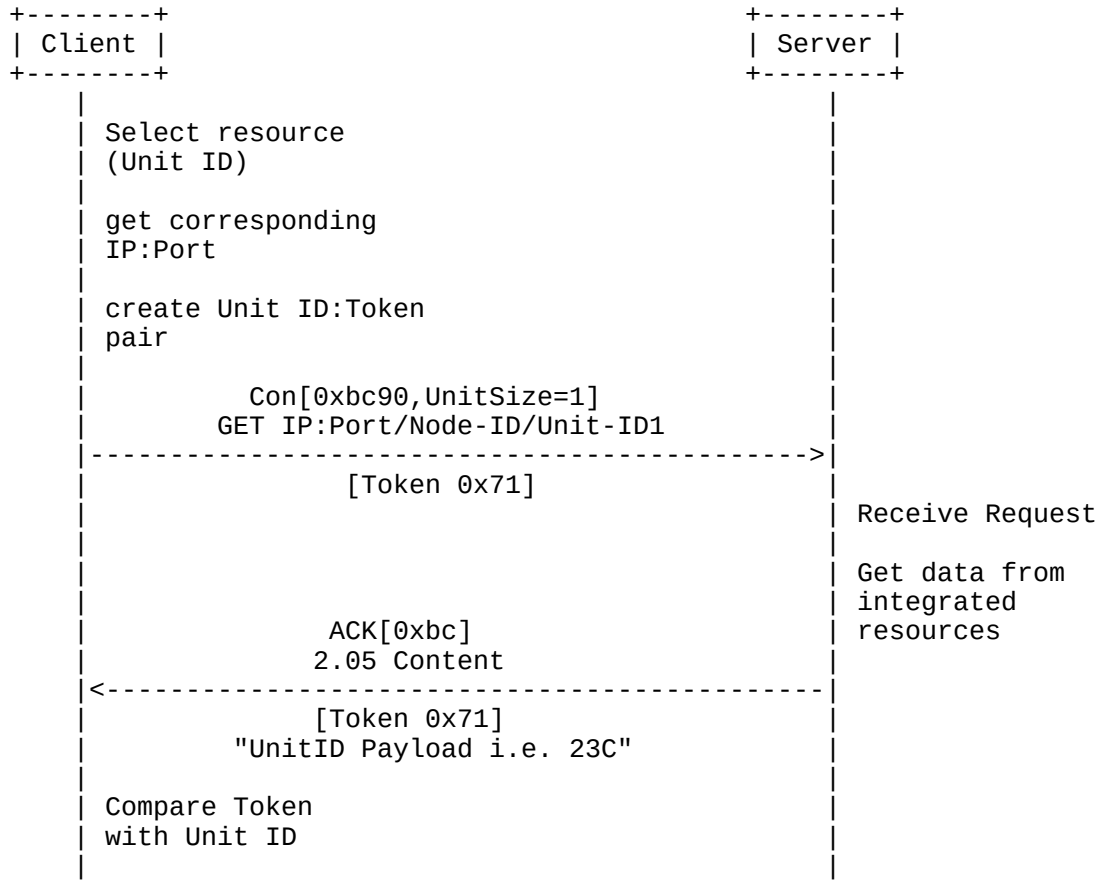

Figure 7: CoAP based client server interaction (single Unit ID)

7.4. CoAP Client Server Interaction (Multiple Units)

Figure 8 shows the interaction among a client and multiple resources i.e. multiple Unit IDs. The example shown in the figure suggests that both Unit IDs belong to a single node but the Unit IDs may also belong to more than one CoAP nodes. As mentioned previously, the client performs lookup on the RD for a specific resource type and gets the list of all the resource IDs (nodes ID and Unit ID) registered with the RD. The following figure shows the process of

client choosing to interact with multiple unit resources (integrated resources) from the list provided by the RD.

Once the client selects the resources' complete URI i.e. Node IP address, Port number and Unit IDs for communication, the client creates and stores the Unit ID, Token pairs.

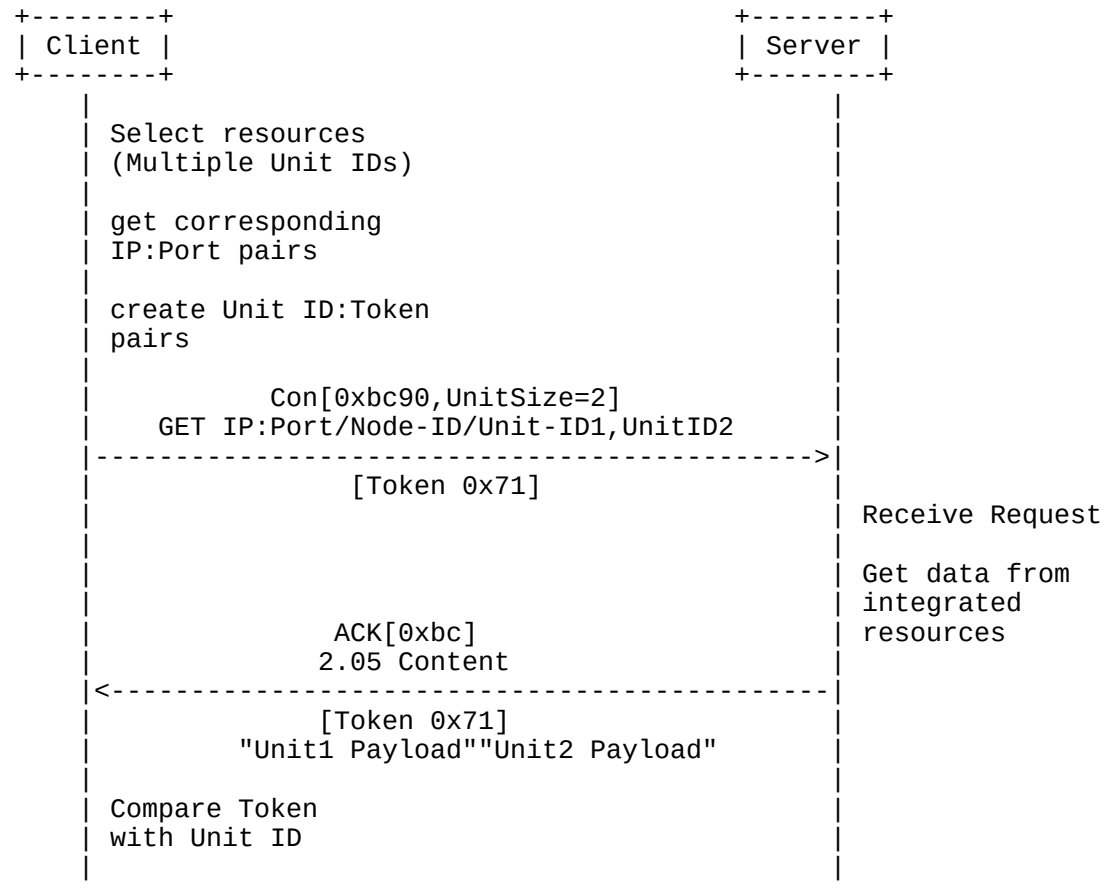

Figure 8: CoAP based client server interaction (Endpoint multiple Unit ID

Here the Token means the CoAP token sent with a normal GET request. The client then sends a GET request to the integrated resources belonging to one or more nodes using the complete URIs (Node IP address, Port number, Node ID, Unit IDs). The GET request with multiple Unit IDs also has the Unit size parameter, mentioning the

number of integrated resources from which the client requests data. The node (CoAP Server), checks the request's validity and responds back to the client with an ACK, consisting of the Token and data from the integrated resources. The client checks the source of the data by comparing the Token of the ACK with the stored Unit ID, Token pairs.

8. Security Considerations

TBD.

9. IANA Considerations

TBD

10. References

10.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC5988] Nottingham, M., "Web Linking", RFC 5988, October 2010.

[RFC6690] Shelby, Z., "Constrained RESTful Environments (CoRE) Link Format", RFC 6690, August 2012.

[RFC7252] Shelby, Z., Hartke, K., and C. Bormann, "The Constrained Application Protocol (CoAP)", RFC 7252, June 2014.

10.2. Informative References

[I-D.ietf-coap-group] Rahman, A. and E. Dijk, "Group Communication for CoAP", ID draft-ietf-core-groupcomm-19, June 2014.

[I-D.ietf-core-rd] Shelby, Z., Bormann, C., and S. Krco, "CoRE Resource Directory", ID draft-ietf-core-resource-directory-01 , December 2013.

[I-D.li-coap-nodeid] Li, K. and G. Wei, "CoAP Option Extension: NodeId", ID draft-li-core-coap-node-id-option-01 , June 2014.

Authors' Addresses

Yong-Geun Hong
ETRI
218 Gajeong-ro Yuseung-Gu
Daejeon 305-700
Korea

Phone: +82 42 860 6557
Email: yghong@etri.re.kr

Younghwan Choi
ETRI
218 Gajeong-ro Yuseung-Gu
Daejeon 305-700
Korea

Phone: +82 42 860 1429
Email: yhc@etri.re.kr

DoHyeun Kim
Jeju Nat. Univ.
Jeju
Korea

Phone: +82 64 754 3658
Email: kimdh@jejunu.ac.kr

Mohammad Sohail Khan
Jeju Nat. Univ.
Jeju
Korea

Phone: +82 64 754 3658
Email: sohail.khan@nwfpuet.edu.pk

WENQUAN JIN
Jeju Nat. Univ.
Jeju
Korea

Phone: +82 64 754 3658
Email: pluskmk12@live.com