ANI Applied in IoT Network Management
draft-rfmesh-anima-iot-management-01

Abstract

This document describes an IoT scenario where ACP and GRASP is suitable to act as a network management channel and a lightweight and extensible network management protocol. Relevant GRASP extension and options are also specified to fulfill the requirements of the scenario.

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 https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on March 30, 2019.

Copyright Notice

Copyright (c) 2018 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal Provisions Relating to IETF Documents (https://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.
1. Introduction

When Anima ANI [I-D.ietf-anima-reference-model] was designed, IoT
scenarios were under consideration. For example, one big reason of
introducing CBOR encoding [RFC7049] in GRASP [I-D.ietf-anima-grasp]
and choosing CoAP [RFC7252] for secure bootstrapping
[I-D.ietf-anima-grasp] is for the efficiency of transporting packets
over lossy IoT networks.

This document discusses applying GRASP and ACP into a specific IoT
scenario for some network management functions. The characteristics
of the scenario is:

- Low-power wireless field area network dedicated for industrial
  usage (e.g. 6LoWPAN-based electronic metering network).
- The topology is mesh. It is natural for a wireless local network.
- IPv6 addressing, which is beneficial for auto-configuration
- L3 routing is enabled (e.g. RPL).
Nodes are extremely resource constrained. (E.g., one typical hardware model only has 128Kbytes RAM and 512Kbytes ROM.)

Gateway is normally a resource rich device, which acts as a management server to the nodes.

Normally nodes don’t need to communicate with any other entities beyond the gateway.

However, some of the ANI designs are not specifically optimized for IoT scenarios:

Most of the GRASP messages (except M_Discovery and M_Flood) are over TCP, which is considered as a heavy burden on radio resources for many IoT LLNs.

Since GRASP is based on TCP, it lacks reliable transport and fragmentation mechanisms by itself.

VRF-based ACP is not applicable for most of the small IoT devices.

This document discusses choosing GRASP as the management protocol over the other two candidates, which are IETF Core technologies and OMA LWM2M technologies. And also discusses a potential lightweight ACP.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119] when they appear in ALL CAPS. When these words are not in ALL CAPS (such as "should" or "Should"), they have their usual English meanings, and are not to be interpreted as [RFC2119] key words.

This document uses the key words defined in [RFC7575].

The following additional terms are used throughout this document:

- IoT: Internet of Things
- BR: Bord Router
- CMD: Command
3. Scenario Description

As Fig 1 depicted, the BR is the root of the wireless network and act as a management server. Each node connects to the BR.

4. Why Choose GRASP as Management Protocol

4.1. Candidate Technologies

4.1.1. IETF Core

Some IoT network management standardization work has been initialed in the IETF Core working group. [I-D.ietf-core-comi] describes a network management interface for constrained devices and networks,
called CoAP Management Interface (CoMI), which is used to access data resources specified in YANG, or SMIv2 converted to YANG; relevant YANG library for CoMI server [I-D.veillette-core-yang-library] and CBOR encoding of data modeled with YANG [I-D.ietf-core-yang-cbor] are also defined. In a nutshell, these work items can be considered as some adaption and optimization of Netconf/YANG technologies for IoT environment.

Netconf/YANG mechanisms are capable of manipulating data organized in a sophisticated tree structure. These capabilities are necessary and powerful in managing various device configurations, especially for the sophisticated devices such as router. However, they might be too heavy for an extremely resource constrained device as described above. There is neither enough space for storing the programs in ROM, nor running the codes in RAM.

4.1.2. OMA LWM2M

OMA had issued the LWM2M specification, which is also designed for IoT network management. LWM2M also chooses CoAP as the management protocol, but it doesn’t choose YANG for data model, rather, it defined some OMA Objects.

OMA objects less complete than YANG modeled data; the objects are flat rather than being organized as a tree structure. But OMA objects contain also some advanced features such as access control of each object. Plus the CoAP implementation, the LWM2M solution is still not ideal for the targeted scenarios in terms of ROM/RAM occupation.

4.2. Suitability of GRASP

According to Section 6.1, most of the IoT commands are more like "Signallings" rather than traditional "Configurations". It is reasonable because the IoT nodes need to auto-configure themselves as much as possible to gain maximum efficiency. Relying on a centralized server configuring each node is a big challenge to the lossy wireless links and might probably cause significant delay of deployment.

Thus, we might need a different approach to consider IoT management than just simply re-using Netconf/YANG in a different context (e.g. CoAP).

5. GRASP Extention

This section discusses potential GRASP extention to fulfill the IoT management requirements.
5.1. GRASP over UDP

Since TCP requires three times handshake, which would consume too much radio resource, thus it is not acceptable in LLNs. Then UDP is needed.

5.2. Reliable Transport

For some critical messages, the sender would need to confirm the receiver had got the message, thus, there needs to be a reliable transport mechanism extended in application layer (GRASP).

5.3. Fragmentation Handling

Since the lack of TCP, GRASP also needs to be enhanced with some a fragmentation mechanism.

6. IoT Management Options Definition

6.1. IoT Management Signallings

This section describes a set of IoT network management commands. These commands are based on a real commercial implementation, however, they are general network management functions that not coupled with any specific services. Thus, these command could be considered as a representative of the general requirements of similar scenarios.

1. NETWORK_HEARTBEAT
   a. BR sends heartbeat to node, every node relay to forward, ACK is optional.
   b. node can send the ACK if needed.

2. NETWORK_DISMISS
   a. CMD from BR to Node: No Options are associated with this CMD. This CMD will be sent in broadcast mode.

3. NODE_REMOVE
   a. CMD from BR to Node: the destination IPv6 address will identify the target node to be removed.
   b. ACK from Node to BR.

4. NODE_LEFT_REPORT
a. Parent node sends a command to BR that a node connected to it has left.
b. BR sends ACK to the parent node.

5. NETWORK_PARA_CONFIG
a. CMD from BR to Node. BR send RF config to every node, based on broadcast relay, ACK is optional.

6. NODE_STATUS
a. Request.
b. Response.

7. NODE_STATISTICS
a. Request.
b. Response.

8. NODE_LOG
a. Request.
b. Response.

9. NODE_RESET
a. first response then reset, when node received this message.

(Editor’s Note: More commands to be extended.)

6.2. GRASP Options

We propose to define three Options as the following. Each of the above mentioned IoT management signallings could be fit into one of the three options as different elements.

- IoT Node Status Reporting. (Details TBD.)
- Management Commands to IoT Nodes. (Details TBD.)
- IoT Network/Node Configuration. (Details TBD.)
7. Lightweight ACP

TBD.

8. Security

TBD.

9. IANA Considerations

TBD.

10. Acknowledgements

Some technical design work was contributed by Shoushou Ren. Relative implementation experience was shared by Zongxin Dou, Wanhong Wang and Haiyan Mao.

Valuable comments were received from Delei Yu, Sheng Jiang and Chuang Wang.

This document was produced using the xml2rfc tool [RFC2629].

11. References

11.1. Normative References

[I-D.ietf-anima-grasp]


11.2. Informative References

[I-D.ietf-anima-autonomic-control-plane]

[I-D.ietf-anima-bootstrapping-keyinfra]

[I-D.ietf-anima-reference-model]

[I-D.ietf-core-comi]

[I-D.ietf-core-sid]

[I-D.ietf-core-yang-cbor]

[I-D.veillette-core-yang-library]


Authors' Addresses

Bing Liu (editor)
Huawei Technologies
Q14, Huawei Campus
No.156 Beiqing Road
Hai-Dian District, Beijing 100095
P.R. China

Email: leo.liubing@huawei.com

Yuefeng Wu
Huawei Technologies
N8, Huawei Campus
No. 101 Ruanjian Road
Yu-Hua-Tai District, Nanjing 210000
P.R. China

Email: wuyuefeng@huawei.com