

CORE WG
Internet-Draft
Intended status: Informational
Expires: April 24, 2013

A. Rahman
InterDigital Communications, LLC
T. Fossati
KoanLogic
S. Loreto
Ericsson
M. Vial
Schneider-Electric
October 21, 2012

Sleepy Devices in CoAP - Problem Statement
draft-rahman-core-sleepy-problem-statement-01

Abstract

This document analyzes the CoAP protocol issues related to sleeping devices. The only goal of this document is to trigger discussions in the CORE WG so that all relevant considerations for sleeping devices are taken into account when designing CoAP.

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/>.

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 April 24, 2013.

Copyright Notice

Copyright (c) 2012 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 (<http://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.

Table of Contents

- 1. Terminology and Conventions 3
- 2. What is a Sleeping Device? 4
 - 2.1. Sleeping behaviors. 4
 - 2.2. Different Sleep Modes 5
 - 2.2.1. Always-On 5
 - 2.2.2. Intermittent Presence 5
- 3. Assumptions 5
- 4. Objectives 6
- 5. Acknowledgements 6
- 6. IANA Considerations 6
- 7. Security Considerations 7
- 8. References 7
 - 8.1. Normative References 7
 - 8.2. Informative References 7
- Authors' Addresses 7

1. Terminology and Conventions

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 [RFC 2119](#) [[RFC2119](#)].

This document assumes readers are familiar with the terms and concepts that are used in [[I-D.ietf-core-coap](#)], [[I-D.ietf-core-link-format](#)], [[I-D.ietf-core-observe](#)], [[I-D.shelby-core-resource-directory](#)].

In addition, this document defines the following terminology:

Sleeping End Point (SEP): is a special kind of CoAP enabled device that spends a large amount of its lifetime disconnected from the network, mainly to save power, or just because it is downright unable to store the energy required for its functioning. It nonetheless owns and hosts a set of resources, and needs to make them available to the other participants in the same constrained RESTful environment. In this respect it has to devise and implement the mechanisms that allows to work around its limitation, and let its resources be accessible as if it were an usual, always connected, CoAP server.

Resource Delegation: is the transfer of control over the handling of a resource from an endpoint (the owner) to another (the deputy), without the actual ownership being relinquished.

The retention of ownership implies two things: first, that a genuine resource delegation cannot be recursive, and second, that it must always be entirely reversible, at any time the owning endpoint deems appropriate.

A Resource Delegation mechanism may comprise the transfer of the following information from the owner to the deputy endpoint:

- (a) complete or partial namespace,
- (b) one or more representations of the resource,
- (c) associated metadata,
- (d) allowed methods,
- (e) access control information,
- (f) temporal bounds of the delegation.

Said mechanism may also provide authentication to the parties involved in the delegation process.

2. What is a Sleeping Device?

A Sleeping device is a device able to cut power to unneeded subsystems and so significantly reduce battery consumption. Some Sleeping devices only cut power to the radio system while continuing to run normally the other ones. Other Sleeping devices are even more energy efficient being able to save the machine state in the RAM memory, putting the RAM into a minimum power state, and cutting power to all the other subsystems. Finally other Sleeping device are able to save the machine state on an hard disk and completely switching off themselves.

2.1. Sleeping behaviors.

In this section we discuss different behaviors and scenarios of sleeping nodes. Such behaviors can affect the design of applications (such as CoAP) and network topologies (such as proxies and caching).

Sleeping nodes can have various sleeping patterns. Sleep patterns can be predictable or totally unpredictable. For example, some nodes sleep at a fixed interval or upon certain triggers. Some nodes may sleep at irregular time intervals, or switch to sleep mode spontaneously. Some nodes stay in sleep mode and only wake up upon certain event triggers. A network may thus not be well aware of the sleeping state of a node at a given time.

The duration of sleeping mode also varies largely, possibly from a few seconds to days. From the perspective of applications, it may not be affected by the sleeping period if it is very short. For example, a HTTP/TCP connection may still work (even if sub-optimally) if the sleep cycle is much shorter than the TCP retransmission timer. In contrast, if the sleeping period of a node exceeds a certain threshold it can impact an application. This threshold however, can be difficult to predict and often can vary from device to device and network to network. For example, this threshold can be very dependent on the topology of a constrained network especially for the case where a multi-hop path consists of multiple sleeping nodes. For this case, the cumulative effect of multiple sleeping nodes must be considered.

The network topology also affects how to handle sleeping nodes. For example, in a star shaped network, a proxy node (assuming to be not a sleeping node) can cache for the sleeping nodes within the network in a centralized manner. However, in a P2P or mesh network, especially when multi-hops are involved, caching can be difficult and delivering of messages can be largely delayed due to nodes' sleeping cycles. In this case distributed proxying and caching at intermediate nodes within the network (rather than just a single node such as the border

node or sink) may make sense, if intermediate nodes are not sleeping nodes and have adequate resources to support caching.

2.2. Different Sleep Modes

2.2.1. Always-On

Any sleep is so short that it is invisible to L3 and upper which gives the illusion of the sleepy node of being always on => usual Server Model can be efficiently used.

2.2.2. Intermittent Presence

Long and possibly non pre-determined sleep periods (more than 1 sec, but typically in the order of minutes or hours) => Server Model not working anymore. SEP state must be handled by other mechanisms.

3. Assumptions

The characteristics of SEPs varies widely. Some may be cheap, rudimentary widgets with very limited computational and storage capabilities; other can be more functional devices yet in need to save energy since they have to be in operation for a long period while battery powered.

This great variance implies that a fair number of often contradictory assumptions must be taken into consideration, and carefully weighted, when designing a comprehensive solution for the problem. For example:

- o Is SEP able to maintain soft state ?
- o Is SEP sleep/awake scheduling predictable ?
- o Is SEP able to handle bidirectional communication ?

Luckily, and by definition, it can be assumed that all the SEPs participating in a CoRE domain share a (realistically limited) subset of the REST principles. At the very least we will assume a SEP understands and implements:

- o the concept of information resource and its representational state;
- o the semantics and syntax of CoAP URIs;

- o the semantics associated with the methods PUT or POST, and DELETE;
- in a way that is conformant with the CoAP protocol. This will provide the common ground on which to build their integration into the hosting CoRE domain.

4. Objectives

The CORE WG aim is to design a solution that, leveraging on the existing CoAP features and its REST architecture, allows SEP devices to be easily and smoothly integrated within any CoRE domain together with the all the other CoAP enabled devices.

The ideal solution should:

- o Make the set of resource owned and hosted by any SEP available to all the other participants, in the same constrained RESTful environment, without making any assumption on the presence of specific or special entities neither on the network topology.
- o Provide the possibility to use Client or Observer Model to access resources owned and hosted by a SEP.
- o Allow the (Secure) delegation of resource handling while retaining ownership.
- o Minimize the configuration needs to bootstrap a SEP within an existing CoRE domain.
- o Maximize the integration with base CoRE Features (i.e. Resource Discovery, Multicast, Observer, Block).
- o Reuse already available CoAP mechanisms as much as possible.

5. Acknowledgements

TBD.

6. IANA Considerations

This memo includes no request to IANA.

7. Security Considerations

TBD. (All drafts are required to have a security considerations section. See [RFC 3552](#) [[RFC3552](#)] for a guide.)

8. References

8.1. Normative References

[I-D.ietf-core-coap]
Shelby, Z., Hartke, K., Bormann, C., and B. Frank,
"Constrained Application Protocol (CoAP)",
[draft-ietf-core-coap-12](#) (work in progress), October 2012.

[I-D.ietf-core-link-format]
Shelby, Z., "CoRE Link Format",
[draft-ietf-core-link-format-14](#) (work in progress),
June 2012.

[I-D.ietf-core-observe]
Hartke, K., "Observing Resources in CoAP",
[draft-ietf-core-observe-06](#) (work in progress),
September 2012.

[I-D.shelby-core-resource-directory]
Shelby, Z., Krco, S., and C. Bormann, "CoRE Resource
Directory", [draft-shelby-core-resource-directory-04](#) (work
in progress), July 2012.

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

8.2. Informative References

[RFC3552] Rescorla, E. and B. Korver, "Guidelines for Writing RFC
Text on Security Considerations", [BCP 72](#), [RFC 3552](#),
July 2003.

Authors' Addresses

Akbar Rahman
InterDigital Communications, LLC
Montreal, Quebec H3A 3G4
Canada

Phone: +1-514-585-0761
Email: akbar.rahman@interdigital.com

Thomas Fossati
KoanLogic

Email: tho@koanlogic.com

Salvatore Loreto
Ericsson
Hirsalantie 11
Jorvas 02420
Finland

Email: Salvatore.Loreto@ericsson.com

Matthieu Vial
Schneider-Electric

Email: matthieu.vial@schneider-electric.com

