

core  
Internet-Draft  
Intended status: Standards Track  
Expires: April 25, 2013

K. Li  
B. Greevenbosch  
Huawei Technologies  
E. Dijk  
Philips Research  
S. Loreto  
Ericsson  
October 22, 2012

**CoAP Option Extension: Patience**  
**draft-li-core-coap-patience-option-01**

**Abstract**

CoAP is a RESTful application protocol for constrained nodes and networks. This specification provides a simple extension for CoAP, the Patience Option. This Option informs a recipient of the preferred time frame for a request or response depending on usage context. In a unicast request, it indicates the patience a client has in waiting for a response. The CoAP server tries to return the response within the specified time frame. In a multicast request, it indicates the patience a server should have in sending its response. The recipient would then try to randomly delay its response within the time frame that the requester indicated or computed by the recipient itself. In a CoAP observe notification, it indicates the patience an observer should have in both waiting for a subsequent notification and in re-establishing an observation relation.

**Note**

Discussion and suggestions for improvement are requested, and should be sent to [core@ietf.org](mailto:core@ietf.org).

**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 25, 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

<a href="#">1.</a>	Introduction . . . . .	<a href="#">4</a>
<a href="#">1.1.</a>	Justification . . . . .	<a href="#">4</a>
<a href="#">1.2.</a>	Terminology . . . . .	<a href="#">5</a>
<a href="#">2.</a>	Patience Option Extension . . . . .	<a href="#">5</a>
<a href="#">2.1.</a>	Patience Option Definition . . . . .	<a href="#">5</a>
<a href="#">2.2.</a>	Using the Patience Option . . . . .	<a href="#">6</a>
<a href="#">2.2.1.</a>	Unicast usage . . . . .	<a href="#">6</a>
<a href="#">2.2.2.</a>	Multicast usage . . . . .	<a href="#">7</a>
<a href="#">2.2.3.</a>	Observe usage . . . . .	<a href="#">7</a>
<a href="#">2.3.</a>	Detection of IP unicast or multicast CoAP request . . . . .	<a href="#">8</a>
<a href="#">3.</a>	Example . . . . .	<a href="#">9</a>
<a href="#">3.1.</a>	Unicast Usage Example . . . . .	<a href="#">9</a>
<a href="#">3.2.</a>	Observe Usage Example . . . . .	<a href="#">9</a>
<a href="#">4.</a>	Security Considerations . . . . .	<a href="#">10</a>
<a href="#">5.</a>	IANA Considerations . . . . .	<a href="#">10</a>
<a href="#">6.</a>	Acknowledgements . . . . .	<a href="#">10</a>
<a href="#">7.</a>	References . . . . .	<a href="#">11</a>
<a href="#">7.1.</a>	Normative References . . . . .	<a href="#">11</a>
<a href="#">7.2.</a>	Informative References . . . . .	<a href="#">11</a>
	Authors' Addresses . . . . .	<a href="#">11</a>



## **1. Introduction**

This specification adds a new option Patience to CoAP [[I-D.ietf-core-coap](#)]. The main purpose is for the requester to inform the recipient of the preferred time frame for a response. In the unicast request case, it is used to indicate the patience it has in waiting for a response. It then indicates "a response is most useful within the specified time frame". In the multicast request case, it indicates the patience that a server should have in sending a response. In other words, it indicates "if possible please delay your response by a randomly chosen time within the specified time frame". A second purpose is for use by a server when sending CoAP observe [[I-D.ietf-core-observe](#)] notifications, to indicate the maximum time an observer should wait (i.e. patience of the observer) before starting any observation relationship recovery.

### **1.1. Justification**

In the unicast case, it is useful for the requester (client) to indicate that the response is required to be returned within a certain amount of time. For example, the requester could require a response within 2 seconds, otherwise the response is not of interest anymore. With this indication of the patience for a response, the requester knows how long it should wait for the response, and it needs to keep the state of the request only for the indicated time. After this period, the request will be given up. It can avoid that the recipient wastes resources by sending a response which already exceeds the set patience timeout of the requester.

In the multicast case, if a server does decide to respond to a multicast request, it should not respond immediately. Instead, it should pick a duration for the period of time during which it intends to respond. For purposes of this exposition, we call the length of this period the Leisure. In [[I-D.ietf-core-observe](#)], how to compute the a rough lower bound for Leisure is specified, and the DEFAULT\_LEISURE is also defined. A Patience option, if present, can be used as an upper bound for the Leisure.

TODO: If the indication of client patience, which is currently only defined for unicast CoAP requests, would also be required for multicast CoAP requests, an alternative to the solution provided in this document is to define two types of Patience Option to be able to express client patience and server patience separately.

In an observe scenario, it is useful for a server to indicate to an observer that, after the period of time in the Max-Age option has expired, a new notification will be sent within the time interval indicated by the Patience option. The server may use this to send



notifications with a dithered delay i.e. randomly chosen within the Patience-specified time interval, when there are many CoAP clients simultaneously observing a resource on the server, avoiding network congestion issues. Another use is for the server to delay sending a new notification because e.g. the resource has not changed. The observer in this case can assume that the server will do its best to deliver a notification at least before the Patience time interval runs out.

If the Patience option is combined with Observe option in a request, currently it indicates the maximum time an observer is prepared to wait for an initial notification.

## 1.2. 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 [\[RFC2119\]](#).

## 2. Patience Option Extension

### 2.1. Patience Option Definition

Type	C/E	Name	Data type	Length	Default
20	E	Patience	pseudo-FP	1 B	(none)

The value carried in the Patience Option is in a specific format resembling a pseudo-Floating Point value (as in [\[I-D.bormann-coap-misc\]](#) [Appendix B.2](#)):

```

0
0 1 2 3 4 5 6 7
+--+--+--+--+--+
|      T      | TX|
+--+--+--+--+--+

```

T = Time

TX = Time Exponent

where the patience time is calculated as:

Patience time =  $2^{(TX * 4 + 3)} * T$





The value of the Patience option is calculated in milliseconds or alternatively mibiseconds ( $1/1024s$ ) if this would ease numerical operations on above values on constrained platforms. The minimum non-zero patience time is 8ms, when  $TX=0$ ,  $T=1$  and a milliseconds time base is used. The maximum patience time is then 2064384ms, around 34 minutes, when  $TX=3$  and  $T=63$ .

## **2.2. Using the Patience Option**

The semantics of the Patience Option depends on its usage context, as detailed in below sections.

### **2.2.1. Unicast usage**

In the unicast case, this option is used by a CoAP client to indicate the maximum time a requester is prepared to wait for a response.

The requester adds the Patience option to any request for which it is prepared to wait for a response. The requester sets the option to the maximum time that it is prepared to wait.

The Patience option applies to both piggy-backed response and separate response. For a separate response, the patience applies to the actual response after the ACK. ACK should be sent immediately upon receipt of the CON message.

TBD: In case a requester retransmits a request, the Patience Option value MAY be decreased by an amount of time equivalent to the time since the previous transmission attempt. In case a requester did not receive an ACK to a confirmable request and a time interval of at least the interval indicated in the Patience Option of the request has passed, the requester SHOULD give up the request.

The recipient interprets this option as the maximum time between receipt of the complete request and the time that it begins sending the response. The requester will observe a longer time interval between request and response, as network transit and processing by proxies add delays. If timing is critical, the requester SHOULD consider the possible delays and choose the value for the option accordingly.

The recipient MAY apply a lower value to the patience timeout based on local policy. A recipient MAY choose to take longer to produce a response, at the risk that the requester is no longer able to use the response.

In case that the CoAP message is transmitted through a proxy, the Proxy MAY reduce the value of a Patience option based on a local



policy (e.g. to consider the maximum time that an idle connection is kept open by a local NAT or Firewall). A Proxy MAY add a Patience option if none is present. The value in the Patience option MUST NOT be increased or removed.

If the requester does not receive a response within the indicated response time, the requester SHOULD consider the request as failed. If the recipient can't provide a response within the required time, the recipient SHOULD discard the request.

This option is "elective". It MUST NOT occur more than once.

### **2.2.2. Multicast usage**

In the multicast case, Leisure is defined in [[I-D.ietf-core-coap](#)] to work as a duration for the period of time during which a server intends to respond to a multicast request. The Patience option in a CoAP request can be used as an upper bound for the Leisure.

How to use Leisure is defined in [[I-D.ietf-core-coap](#)].

This option is "elective". It MUST NOT occur more than once.

### **2.2.3. Observe usage**

In a CoAP observe [[I-D.ietf-core-observe](#)] scenario, the Patience Option MAY be used in a notification to indicate the maximum time an observer should wait before starting any observation relationship recovery.

The Max-Age Option indicates the maximum time a response (notification) may be cached before it MUST be considered stale. The Max-Age Option of a notification is usually set to a value that estimates when the server will send the next notification. However, in the case the value has not changed, the server can decide not to send a new notification, possibly confusing the observer. It is quite difficult for an observer to discriminate the situation that it has not received a new notification because the value has not changed from situations where the server has lost its state, or for some reason has given up on notification delivery.

The Patience Option in a notification is used to indicate the maximum time a server will try to reach the client before giving up. This is to save the client some effort in re-establishing observation relationships each time max-age is reached. This option is also useful to give a server the time to send out the notifications, in case there are many CoAP clients observing simultaneously a resource, while avoiding network congestion issues.



The server adds the Patience option to any notification related to an observation relationship from which it wants delay an observation refresh request made by the observer. The server sets the option to the maximum time that it is prepared to spend to reach the observer before giving up.

The observer interprets this option as the minimum time between the expiration of a notification (as indicated by its Max-Age Option value) and the moment it MAY start an observation relationship recovery action with the server.

If the observer does not receive a response within the indicated time interval, the observer SHOULD attempt to re-establish the observation relationship with the server if it is still interested in observing the particular resource.

This option is "elective". It MUST NOT occur more than once.

### **2.3. Detection of IP unicast or multicast CoAP request**

A single Patience Option, used to indicate potentially either client patience (in the IP unicast case) or server patience (in the IP multicast case), requires that a CoAP server is able to distinguish between IP unicast and multicast requests. If there exist commonly used IP stacks that do not offer such functionality [to be checked], requiring servers to be able to make the unicast/multicast distinction seems unwise and limits the applicability of the Patience Option.

Approaches for a CoAP server to detect unicast versus multicast requests may include:

- 1) CoAP server application opens a specific socket and sets IP multicast reception using the POSIX setsockopt function [to be verified if IP unicast traffic also is received in this case, or not].
- 2) CoAP server checks the IP destination address of incoming packets. If this has the FF00::/8 IPv6 prefix, then it's treated as multicast otherwise unicast [to be verified if IP stack APIs allow to get IP destination].
- 3) Receiving CoAP multicast requests always occurs on a different port than the standard CoAP port. For example, similar to coaps:// that uses a different port than coap://, a scheme coapm:// on a different port may be defined for multicast requests.



### 3. Example

#### 3.1. Unicast Usage Example

This section gives a short example with a message flow that illustrates the use of the Patience option in a GET request.

This example (Figure 1) shows that the requester wants to get a response within 3200 milliseconds, when T=25, TX=1.

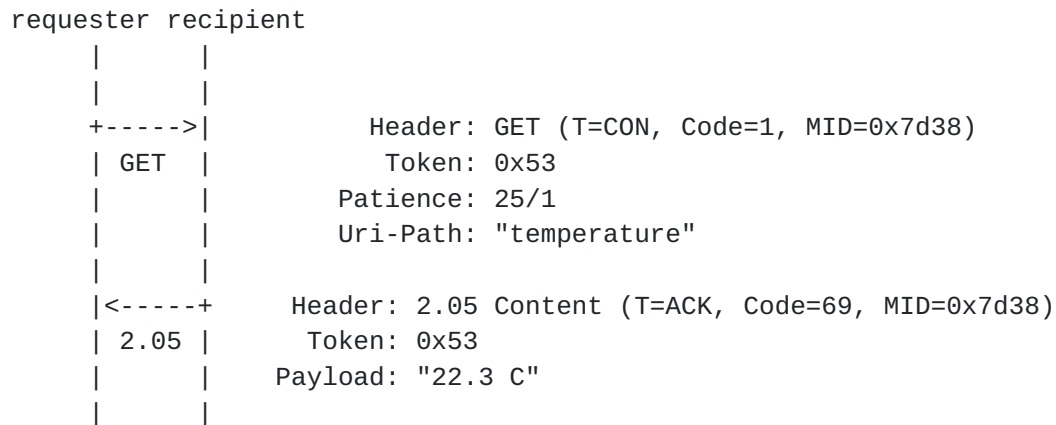


Figure 1: Patience Option in a unicast request

#### 3.2. Observe Usage Example

This section gives a short example with a message flow that illustrates the use of the Patience option in an Observe notification.

This example (Figure 2) shows that the server wants the observer to wait 819 seconds (T=25, TX=3) before starting any observation relationship recovery, even though the Max-Age of the temperature value notification is only 120 seconds.





```

Observer Server
|             |
|             |
+----->|      Header: GET (T=CON, Code=1, MID=0x7d38)
| GET  |      Token: 0x53
|       |      Observe: 0
|       |      Uri-Path: "temperature"
|       |
|<-----+      Header: 2.05 Content (T=ACK, Code=69, MID=0x7d38)
| 2.05 |      Token: 0x53
|       |      Max-Age: 120
|       |      Patience: 25/3
|       |      Payload: "22.3 C"
|       |
|       |

```

Figure 2: Patience Option in an observe notification

#### 4. Security Considerations

This presents no security considerations beyond those in [section 10](#) of the base CoAP specification [[I-D.ietf-core-coap](#)].

#### 5. IANA Considerations

The IANA is requested to add the following "CoAP Option Numbers" entry as per Section 11.2 of [[I-D.ietf-core-coap](#)].

Number	Name	Reference
20	Patience	<a href="#">Section 2</a>

#### 6. Acknowledgements

The authors of this draft would like to thank the participants of the email discussion on this issue. Thanks to Carsten Bormann, Peter Bigot, Barry Leiba, Linyi Tian for the reviews and discussions.

#### 7. References



### **7.1. 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-observe]  
Hartke, K., "Observing Resources in CoAP",  
[draft-ietf-core-observe-06](#) (work in progress),  
September 2012.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate  
Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.

### **7.2. Informative References**

- [I-D.bormann-coap-misc]  
Bormann, C. and K. Hartke, "Miscellaneous additions to  
CoAP", [draft-bormann-coap-misc-21](#) (work in progress),  
October 2012.

#### **Authors' Addresses**

Kepeng Li  
Huawei Technologies  
Huawei Base, Bantian, Longgang District  
Shenzhen, Guangdong 518129  
P. R. China

Phone: +86-755-28974289  
Email: [likepeng@huawei.com](mailto:likepeng@huawei.com)

Bert Greevenbosch  
Huawei Technologies  
Huawei Base, Bantian, Longgang District  
Shenzhen, Guangdong 518129  
P. R. China

Phone: +86-755-28978088  
Email: [bert.greevenbosch@huawei.com](mailto:bert.greevenbosch@huawei.com)



Esko Dijk  
Philips Research  
High Tech Campus 34  
Eindhoven  
The Netherlands

Email: [esko.dijk@philips.com](mailto:esko.dijk@philips.com)

Salvatore Loreto  
Ericsson  
Hirsalantie 11  
Jorvas 02420  
Finland

Email: [salvatore.loreto@ericsson.com](mailto:salvatore.loreto@ericsson.com)

