

**CoRE Simple Server Discovery**  
**draft-bormann-core-simple-server-discovery-00**

Abstract

CoRE defines a mechanism for resource discovery based on Web linking. Many applications also need a simple form of discovery for the servers carrying these resources. This specification shows a simple way to extend the link-based resource discovery into a basic form of server discovery.

The current version -00 of this document is just an initial draft that is intended to spark discussion.

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 September 8, 2011.

Copyright Notice

Copyright (c) 2011 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.](#) Introduction . . . . . [3](#)
- [1.1.](#) Terminology . . . . . [3](#)
- [2.](#) Discovery Servers . . . . . [4](#)
- [3.](#) CoAP Server Discovery . . . . . [5](#)
- [4.](#) Finding a Candidate CoAP Server Discovery Server . . . . . [6](#)
- [5.](#) IANA Considerations . . . . . [7](#)
- [6.](#) Security Considerations . . . . . [8](#)
- [7.](#) Acknowledgements . . . . . [9](#)
- [8.](#) References . . . . . [10](#)
- [8.1.](#) Normative References . . . . . [10](#)
- [8.2.](#) Informative References . . . . . [10](#)
- Author's Address . . . . . [11](#)

## **1. Introduction**

CoRE defines a mechanism for resource discovery based on Web linking [[RFC5988](#)] [[I-D.ietf-core-link-format](#)]. Many applications also need a simple form of discovery for the servers carrying these resources.

More sophisticated CoRE server discovery mechanisms have been proposed [[I-D.brandt-coap-subnet-discovery](#)]. The present specification is not intended as a competing protocol but shows a very simple way to extend the link-based resource discovery into a basic form of server discovery. It is an open question whether different applications need different discovery solutions or whether there can be a "scalable" solution that covers both simple and complex scenarios.

The protocol as designed here has been prototyped in the SAHARA project at TZI in just a few lines of code. The current version -00 of this document serves as an initial draft that is intended to spark discussion. Not all aspects of the protocol as specified are part of the current prototype. We expect to update the specification both based on WG feedback and as we gain experience with the prototype.

### **1.1. 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 [RFC 2119](#). (See [[RFC2119](#)].)

The term "byte" is used in its now customary sense as a synonym for "octet".

The terminology from [[I-D.ietf-core-coap](#)] applies.

In addition:

CoAP Server Discovery: This protocol.

CoAP Server Discovery Server (CSDS): A server for this protocol, which interacts with other CoAP servers, collects resource discovery information from them and integrates it into larger resource discovery information sets.

Candidate CSDS: An IP address that might or might not be useful for conversion to a CSDS URI.

## **2. Discovery Servers**

This specification defines a simple form of server discovery that makes use of CoAP Server Discovery Servers (CSDS), which are addressed simply using the CoAP protocol [[I-D.ietf-core-coap](#)].

The assumption is that there is a way to find one or more CSDSs (see [Section 4](#) for a number of such ways). New CoAP servers that want to provide discoverable services can make themselves known at the CSDSs. CoAP clients can ask the CSDSs for a resource directory in the usual way, which will include both information about the discovery server's own resources and information about other servers that made themselves known to the discovery servers.

### **3. CoAP Server Discovery**

Simple CoAP Server Discovery makes use of a simple mapping from a server's IP address to a default discovery URI: The default discovery URI is created from the server IP address, the CoAP default port [[I-D.ietf-core-coap](#)], and the absolute path `"/.well-known/core"` [[I-D.ietf-core-link-format](#)].

A CoAP server that wants to make itself discoverable occasionally sends a POST request to the default discovery URI of any Candidate CSDS that it finds.

The body of the POST request is either

- o empty, in which case the CoAP Server Discovery Server is encouraged by this POST request to perform GET requests at the requesting server's default discovery URI.

or

- o a link-format document, which indicates the specific services that the requesting server wants to make known to the CSDS.

The CSDS integrates the information it received this way into its resource directory. It MAY make the information available to further CSDSs, if it can ensure that a loop does not form. The protocol used between CSDSs to ensure loop-free operation is outside the scope of this document.

#### **4. Finding a Candidate CoAP Server Discovery Server**

CoAP servers that want to contact a CSDS can obtain candidate IP addresses for such servers (Candidate CSDS) in a number of ways.

In a 6LoWPAN, good candidates can be taken from:

- o specific static configuration (e.g., anycast addresses), if any,
- o the ABRO option of 6LoWPAN-ND [[I-D.ietf-6lowpan-nd](#)],
- o other ND options that happen to point to servers (such as RDNSS),
- o DHCPv6 options that might be defined later.

In networks with more inexpensive use of multicast, the Candidate CSDS may be a well-known multicast address, i.e. CSDS are found by simply sending POST requests to that well-known multicast address (details TBD).

As some of these sources are just (more or less educated) guesses, CoAP servers MUST make use of any error messages to very strictly rate-limit requests to Candidate CSDSs that don't work out. E.g., an ICMP Destination Unreachable message (and, in particular, the port unreachable code for this message) may indicate the lack of a CoAP server on the candidate host, or a CoAP error response code such as 4.05 "Method Not Allowed" may indicate unwillingness of a CoAP server to act as a CSDS.

## **5. IANA Considerations**

This document has no actions for IANA.

## **6. Security Considerations**

(None so far; this section will certainly grow as additional security considerations beyond those listed in the base specifications become known.)



## **7. Acknowledgements**

The concept for this document was inspired by Zach Shelby et al.'s CoAP discovery node that was available in various CoAP interop events. The current implementation was performed by the students of the SAHARA project, including Bengt Kohrt, Julian Kornberger, Henning Mueller, and Christian Thedieck. Philip Nguyen read an early draft of this document (but all errors are mine). Anders Brandt's draft [[I-D.brandt-coap-subnet-discovery](#)] is a fine piece of work and certainly motivated me to finally write this up.

## **8. References**

### **8.1. Normative References**

- [I-D.ietf-6lowpan-nd]  
Shelby, Z., Chakrabarti, S., and E. Nordmark, "Neighbor Discovery Optimization for Low-power and Lossy Networks", [draft-ietf-6lowpan-nd-15](#) (work in progress), December 2010.
- [I-D.ietf-core-coap]  
Shelby, Z., Hartke, K., Bormann, C., and B. Frank, "Constrained Application Protocol (CoAP)", [draft-ietf-core-coap-04](#) (work in progress), January 2011.
- [I-D.ietf-core-link-format]  
Shelby, Z., "CoRE Link Format", [draft-ietf-core-link-format-02](#) (work in progress), December 2010.
- [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.

### **8.2. Informative References**

- [I-D.brandt-coap-subnet-discovery]  
Brandt, A., "Discovery of CoAP servers across subnets", [draft-brandt-coap-subnet-discovery-00](#) (work in progress), March 2011.

Author's Address

Carsten Bormann  
Universitaet Bremen TZI  
Postfach 330440  
Bremen D-28359  
Germany

Phone: +49-421-218-63921  
Fax: +49-421-218-7000  
Email: [cabo@tzi.org](mailto:cabo@tzi.org)