

DTN Research Group	A. McMahon	
Internet-Draft	Trinity College Dublin	
Intended status: Experimental	K. Fall	
Expires: August 31, 2010	Intel Labs, Berkeley	
	February 27, 2010	

[TOC](#)

The Delay Tolerant Networking Endpoint Discovery Protocol draft-mcmahon-dtnrg-dtn-edp-00

Abstract

The Endpoint Discovery Protocol (EDP) can be used by Delay Tolerant Networking (DTN) nodes to report their storage, routing and security capabilities, endpoint identifiers (EIDs), registrations and whether they are willing to be custodians to neighbouring DTN nodes.

Status of this Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/1id-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

This Internet-Draft will expire on August 31, 2010.

Copyright Notice

Copyright (c) 2010 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 BSD License.

Table of Contents

- [1.](#) Introduction
- [2.](#) Requirements Language
- [3.](#) The EDP Registrations
 - [3.1.](#) The Source EID of a Bundle with an EDP payload
 - [3.2.](#) The CLA EID
 - [3.3.](#) EDP Flags for the Primary Block
- [4.](#) The EDP Bundle
 - [4.1.](#) The EDP Report Bundle Format
 - [4.2.](#) Active Registrations.
 - [4.3.](#) The EDP Application Agent.
 - [4.3.1.](#) Bundle Generation and Transmission.
 - [4.3.2.](#) Bundle Reception.
 - [4.3.2.1.](#) Random Response Request.
 - [4.3.2.2.](#) Scheduled Response Request.
 - [4.4.](#) Example: Three DTN Nodes.
 - [4.4.1.](#) CLA EIDs of 'A', 'B' and 'C'
 - [4.4.2.](#) The ARs, CLA EIDs and Preferences of 'C'
 - [4.4.3.](#) 'C' Transmits Bundle containing an EDP ADU
 - [4.4.4.](#) 'A' Uses Service of 'C'
 - [4.4.5.](#) 'B' Uses Service of 'C'

- [5.](#) Acknowledgements
- [6.](#) IANA Considerations
- [7.](#) Security Considerations
- [8.](#) References
- [8.1.](#) Normative References
- [8.2.](#) Informative References
- [Appendix A.](#) DTN Routing Protocols And Algorithms
- [Appendix B.](#) CLA Parameters
- [Appendix C.](#) APIs for EDP
- [Appendix D.](#) The Data Dictionary
- [Appendix E.](#) Discovery Mechanisms
- [§](#) Authors' Addresses

1. Introduction

[TOC](#)

This document describes Version 1 of the Endpoint Discovery Protocol (EDP), EDPv1, designed to be used with the Bundle Protocol (BP) [[RFC5050](#)] ([Scott, K. and S. Burleigh, "Bundle Protocol Specification,"](#)

[November 2007.](#)) within the context of the Delay-Tolerant Networking (DTN) architecture [\[RFC4838\] \(Cerf, V., Burleigh, S., Hooke, A., Torgerson, L., Durst, R., Scott, K., Fall, K., and H. Weiss, "Delay-Tolerant Networking Architecture," April 2007.\)](#).

EDP can be used by a DTN node to discover the presence of DTN nodes wishing to receive EDP Application Data Units (ADUs) on their EDP registrations, and to discover specifically which Convergence Layer Adapters (CLAs) and registrations are of interest to those proximate DTN nodes. A DTN node with an EDP registration is an EDP node.

EDP nodes report a description of their active EID registrations, Convergence Layer Adapters (CLAs) EIDs, storage, routing, security capabilities and whether they are willing to be custodian to proximate EDP nodes.

A DTN node MUST have a unique EDP registration. EDP is envisioned to utilize a (to be agreed upon) mechanism to limit its distribution scope to one DTN hop. Routing protocols (to be agreed upon) are expected to make use of information learned via EDP.

EDP is a DTN application protocol. EDP ADUs are encapsulated in bundles [\[RFC5050\] \(Scott, K. and S. Burleigh, "Bundle Protocol Specification," November 2007.\)](#) that may be secured according to BSP [\[BPsec\] \(Symington, S., Farrell, S., Weiss, H., and P. Lovell, "Bundle Security Protocol Specification," February 2010.\)](#).

This document specifies the protocol, block formats, and abstract service description for the exchange of EDP ADUs. This document does not address:

*Operations in the CLAs.

*Discovery of EIDs more than one DTN node hop away.

2. Requirements Language

[TOC](#)

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

3. The EDP Registrations

[TOC](#)

Within a DTN node, there is a registration which is the state machine characterizing a given node's membership in a given endpoint. Any number of registrations may be concurrently associated with a given endpoint,

and any number of registrations may be concurrently associated with a given node. An EID is a name, expressed using the general syntax of URIs [RFC3986] (Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax," January 2005.), that identifies a DTN endpoint. EDP uses the "dtn:" scheme [dtnURI] (Fall, K., Burleigh, S., Doria, A., and J. Ott, "The DTN URI Scheme," September 2009.). The EDP EID registration, used by an application for receiving EDP ADUs, has the non-expiring registration "dtn::EDPv1". An EDP registration receives EDP ADUs, which are report bundles generated periodically or on demand and transmitted to the endpoint 'dtn::EDPv1'. Registrations are strings, that might indicate a physical entity or named data/service, used by upper-layer protocols or DTN applications to ask the CLAs to enable and disable reception of ADUs sent to specific EIDs.

3.1. The Source EID of a Bundle with an EDP payload

[TOC](#)

Each DTN node is required to have at least one EID that uniquely identifies it (see section 3.3 of RFC 4838 (Cerf, V., Burleigh, S., Hooke, A., Torgerson, L., Durst, R., Scott, K., Fall, K., and H. Weiss, "Delay-Tolerant Networking Architecture," April 2007.) [RFC4838]). The source EID of an EDP report bundle is any of those registrations.

3.2. The CLA EID

[TOC](#)

The EID of a CLA is a CLA EID as defined by [dtnURI] (Fall, K., Burleigh, S., Doria, A., and J. Ott, "The DTN URI Scheme," September 2009.). CLA EIDs are primarily used for the identification of CLAs and are used by EDP as potential "next hops".

Scheme	Scheme Specific Part (SSP)
dtn:	next:eui-48:00:1c:bf:93:98:5d
dtn:	next:eui-48:00:1b:38:cc:df:ef

Example of a DTN node with two Ethernet CLA EIDs.

Table 1: CLA EID

3.3. EDP Flags for the Primary Block

[TOC](#)

The value encoded in the Bundle Processing Control Flags of the primary block is a string of bits, expressed as a SDNV and is used to invoke selected bundle processing control features. By default, a bundle with a payload that is an EDP ADU has the individual bits of this string of bits set to zero to indicate false for the following flags:

- Custody transfer requested
- Status report request
- Application data unit is an administrative record
- Bundle must not be fragmented
- Destination endpoint is a singleton
- Acknowledgement by application is requested

4. The EDP Bundle

[TOC](#)

The EDP payload block uses the Canonical Bundle Block Format as defined in section 4.5.2. of [RFC 5050 \(Scott, K. and S. Burleigh, "Bundle Protocol Specification," November 2007.\)](#) [RFC5050]. That is, each EDP block is comprised of the following elements:

- Block type code
- Block processing control flags
- Block data length
- Block EID reference count and EID references
- Block-type-specific data fields

A bundle with an EDP payload is uniquely identifiable and all bundle protocol features that rely on bundle identity, such as the Bundle Security Protocol [\[BPsec\] \(Symington, S., Farrell, S., Weiss, H., and P. Lovell, "Bundle Security Protocol Specification," February 2010.\)](#), can be enabled. A bundle with an EDP payload may be fragmented. The 'block contains an EID-reference field' flag in the block processing control flags of the primary block is set to 1 as the EDP block references EID elements in the primary block's dictionary.

4.1. The EDP Report Bundle Format

[TOC](#)

EDP ADUs contained in bundles are called EDP reports. In order to take full advantage of the capabilities of EDPv1, an EDP node MUST be capable of generating a bundle payload block, a record with the below format and of transmitting that bundle to the EDP EID.

```
+-----+-----+-----+-----+
| Block type   | Proc. Flags (*)| Block length(*)| DCF (*)      |
+-----+-----+-----+-----+
|              | AR EID Reference Count (*) |
+-----+-----+-----+-----+
| AR list Ref_scheme_1 (*) | AR list Ref_esp_1 (*) |
+-----+-----+-----+-----+
| AR expiration counters (*) |
+-----+-----+-----+-----+
|              | CLA EID Reference Count (*) |
+-----+-----+-----+-----+
| CLA list Ref_scheme_1 (*) | CLA list Ref_esp_1 (*) |
+-----+-----+-----+-----+
|              | PREF CLA Reference Count (*) |
+-----+-----+-----+-----+
| PREF list Ref_AR_I (*)   | PREF list Ref_CLA_I (*)   |
+-----+-----+-----+-----+
| SC (*)                   | SA (*)                     |
+-----+-----+-----+-----+
|              | Delay (*)                   |
+-----+-----+-----+-----+
```

A [\[RFC5050\]](#) (Scott, K. and S. Burleigh, "Bundle Protocol Specification," November 2007.) Canonical Bundle Block Format

Figure 1: The EDP Report Bundle Format

Notes:

*EDP is a DTN application protocol that uses the canonical bundle block format [\[RFC5050\]](#) (Scott, K. and S. Burleigh, "Bundle Protocol Specification," November 2007.)

() Indicates field contains a Self Delimiting Numerical Value (SDNV) (see Section 4.1. of [RFC 5050](#) (Scott, K. and S. Burleigh, "Bundle Protocol Specification," November 2007.) [RFC5050])

- *Routing protocols and algorithms are identified by EIDs [[dtnURI](#)] ([Fall, K., Burleigh, S., Doria, A., and J. Ott, "The DTN URI Scheme," September 2009.](#)) and have EID registrations (see [Appendix A \(DTN Routing Protocols And Algorithms\)](#)). Active registrations (ARs) have registration references in the EDP payload block.

- *"Block Type" (see section 3.1 of [RFC 5050 \(Scott, K. and S. Burleigh, "Bundle Protocol Specification," November 2007.\)](#) [RFC5050]) is a 1-byte mandatory field that indicates the type of the block. This field contains the value 1 to indicate it is the bundle payload block.

- *"Block Processing Control Flags" (see section 3.1 of [RFC 5050 \(Scott, K. and S. Burleigh, "Bundle Protocol Specification," November 2007.\)](#) [RFC5050]) is a SDNV that contains the BP block processing control flags, an unsigned integer expressed as a SDNV. The individual bits of this integer are set to '1' for the Last block flag, Discard block if it can't be processed flag and the block contains an EID-reference flag. A one octet SDNV is shown here for convenience in representation.

- *"Block Length" (see section 3.1 of of [RFC 5050 \(Scott, K. and S. Burleigh, "Bundle Protocol Specification," November 2007.\)](#) [RFC5050]) is a mandatory SDNV that contains the aggregate length of all remaining fields of the block - which is to say, the length of the bundle's payload. A one octet SDNV is shown here for convenience in representation.

- *The "DCF" (Discovery Control Flags field) in a bundle with an EDP payload is a SDNV; the value encoded in this SDNV is a string of bits used to invoke selected block processing control features. A one octet SDNV is shown here for convenience in representation. The significance of the values in all currently defined positions of this bit string, in order from least significant position in the decoded bit string (labeled '0') to most significant (labeled '6'), are described here.

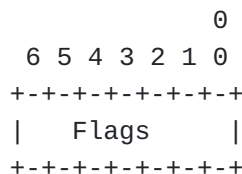


Figure 2: Block Discovery Control Flags Bit Layout

0 - Confirm BSP. If set indicates the Bundle Security Protocol [BPsec] ([Symington, S., Farrell, S., Weiss, H., and P. Lovell, "Bundle Security Protocol Specification," February 2010.](#)) is enabled on the EDP node.

1 - Confirm custodian status. If set indicates an EDP node is prepared to be a custodian to any neighbouring DTN nodes.

2 - Confirm reactive fragmentation. If set indicates reactive fragmentation is enabled. The reactive fragmentation capability is not required to be available in every DTN implementation (See section 3.9) [RFC 5050 \(Scott, K. and S. Burleigh, "Bundle Protocol Specification," November 2007.\)](#) [RFC5050].

3 - Request EDP. If set indicates that a bundle with an EDP payload is requested from the EDP node that receives this bundle.

4 - Random response request. Indicates that the EDP node that receives this bundle is requested to generate a random positive integer uniformly distributed between zero and the value in the delay field of the EDP payload. This integer represents seconds to delay before sending an EDP report in response.

5 - Reserved.

6 - Reserved.

*"AR EID reference count" is an EID reference field consisting of a count of EID active registration references expressed as an SDNV and followed by the EID references themselves. Each EID reference is a pair of SDNVs as described below. A four octet SDNV is shown here for convenience in representation.

*"AR list Ref_scheme_1" Contains the offset of a scheme name in the primary block's dictionary for an active registration. If present, is a SDNV and is therefore variable length. A two-octet SDNV is shown here for convenience in representation.

*"AR list Ref_ssp_1" contains the offset of a scheme-specific part in the primary block's dictionary for an active registration. If present, is a SDNV and is therefore variable length. A two-octet SDNV is shown here for convenience in representation.

*"AR expiration counters" is a list of the registration expiration counters for each active registration. It is an integer that indicates the expiration time of an active registration in seconds

expressed as a SDNV and is therefore variable length. A four octet SDNV is shown here for convenience in representation.

*"CLA EID reference count" is an EID reference field consisting of a count of CLA EID references expressed as an SDNV and followed by the EID references themselves. Each CLA EID reference is a pair of SDNVs as described below. A four octet SDNV is shown here for convenience in representation.

*"CLA list Ref_scheme_1" Contains the offset of a scheme name in the primary block's dictionary for a CLA EID. If present, is a SDNV and is therefore variable length. A two-octet SDNV is shown here for convenience in representation.

*"CLA list Ref_ssp_1" contains the offset of a scheme-specific part in the in the primary block's dictionary for a CLA EID. If present, is a SDNV and is therefore variable length. A two-octet SDNV is shown here for convenience in representation.

*"PREF CLA reference count" is an EID reference field consisting of a count of the preferred CLAs (interfaces between the common bundle protocol and a specific inter-network protocol suite) for each active registration reference expressed as an SDNV and followed by the EID counter references themselves. Each EID counter reference is a pair of SDNVs as described below. A four octet SDNV is shown here for convenience in representation.

*"PREF list Ref_AR_I" is a reference field to the "AR EID reference count" field consisting of an instance of the count of EID active registration references. It is an index into the "AR EID reference count" to identify an AR EID reference. If present, is a SDNV and is therefore variable length. A two-octet SDNV is shown here for convenience in representation.

*"PREF list Ref_CLA_I" is a reference field to the "CLA EID reference count" field consisting an instance of the count of CLA EIDs references, followed by a delimiter ',' and then an integer. The instance of the count of CLA EIDs references is an index into the "CLA EID reference count" to identify and to identify a CLA EID reference. The integer, following the delimiter is set to '1' by default and is an indicator of CLA EID preference (see [Appendix B \(CLA Parameters\)](#)). An integer increase signifies a decrease of preference. An instance of the count of CLA EIDs references and the preference integer may both be set to '0' to signify no preference. If present, is a SDNV and is therefore variable length. A two-octet SDNV is shown here for convenience in representation.

*"SC" is the DTN node storage capacity in bytes expressed as a SDNV and is therefore variable length. A two octet SDNV is shown here for convenience in representation.

*"SA" is the DTN node's available storage capacity in bytes expressed as a SDNV and is therefore variable length. A two octet SDNV is shown here for convenience in representation.

*"Delay" is a positive integer which indicates a delay in seconds. When DCF field bit position 3 is set this field is used to delay transmission of a bundle with an EDP payload report by a random or set amount of seconds. The delayed transmission time is computed by the receiving EDP node by adding the delay integer to the creation time stamp of the received bundle. In the event an EDP node receives a request to transmit a report at some time in the past, the EDP node will not generate a report. If present, is a SDNV and is therefore variable length. A four-octet SDNV is shown here for convenience in representation.

4.2. Active Registrations.

[TOC](#)

All active registrations and CLA EIDs referred to in the blocks of a bundle with an EDP payload are conveyed in the "dictionary" byte array in the bundle's primary block. This array is simply the concatenation of any number of null-terminated scheme names and SSPs. "Endpoint ID references" field in the payload block are used to cite endpoint IDs that are contained in the dictionary.

4.3. The EDP Application Agent.

[TOC](#)

4.3.1. Bundle Generation and Transmission.

[TOC](#)

The EDP application agent is responsible for constructing EDP report ADUs to be transmitted in bundles. By default, bundles with EDP payloads are generated and transmitted periodically every 30 seconds. EDP configuration options can be used to change the default transmission period or to instruct the EDP agent to only transmit in response to another DTN nodes' EDP report. Bundles with EDP payloads can also be generated on demand. In concept, the EDP application agent discharges this responsibility by directing the node's application agent to

construct the record in the payload and request its transmission. The BP application agent constructs the record as is necessary to add a byte array, expressed as a SDNV, to the primary block's dictionary (see [Appendix D \(The Data Dictionary\)](#))

4.3.2. Bundle Reception.

[TOC](#)

The EDP agent is responsible for receiving an EDP ADU on the EDP registration and processing it.

An EDP ADU with the 'Request EDP' flag set in its status field is an 'EDP request'. An EDP agent that receives an 'EDP request' is directed to schedule generation and transmission of an EDP ADU. An EDP ADU will be transmitted at a time equal to that of the creation timestamp plus the delay value indicated in the delay field of the 'EDP request'. When an EDP request is received with the 'Request EDP' flag set, the generated bundle with an EDP payload MUST have the 'Request EDP' flag set to zero.

4.3.2.1. Random Response Request.

[TOC](#)

When an 'EDP request' is received that has the 'Random response request' flag set in its status field, the EDP application agent schedules generation and transmission of an EDP ADU in a delayed fashion. On reception of the 'EDP request' the EDP agent generates a specific delay value, a random positive integer. Generation of the integer is bound by a range indicated by the value specified in the delay field in the 'EDP request' payload. A positive integer, indicating a delay in seconds, is generated if the value is positive. A delay value of zero indicates a response should be generated immediately.

4.3.2.2. Scheduled Response Request.

[TOC](#)

A bundle with an EDP ADU in its payload is transmitted on reception of an 'EDP request' when the delay field is zero (default), as this indicates that there is no delay desired. If the 'Request EDP' flag is set and the 'Random response request' flag is not set, the delay value indicates an absolute delay in seconds. This type of 'EDP request' is used to request a remote EDP application agent to schedule generation and transmission of a bundle with an EDP ADU in its payload at a specific later time.

4.4. Example: Three DTN Nodes.

[TOC](#)

A scenario in which three proximate DTN nodes 'A', 'B' and 'C', identifiable by unique EIDs 'dtn::A', 'dtn::B' and 'dtn::C' is presented. 'C' has three services 'X', 'Y' and 'Z'. Each service has an active EID registration, in the form 'unique EID' '/' 'service'. 'A', 'B' and 'C' publish their services by transmitting bundles with EDP ADUs in the payload to the destination EID 'dtn::EDPv1', of proximate DTN nodes via all CLAs. In this scene the three registrations for 'C' are as follows:

```
dtn::C/X
```

```
dtn::C/Y
```

```
dtn::C/Z
```

4.4.1. CLA EIDs of 'A', 'B' and 'C'

[TOC](#)

Each DTN node is equipped with two Ethernet interfaces and a Bluetooth interface. The CLA EID that is used to identify each CLA includes a CLA type and an CLA instance identifier. In this scene, the CLA type for the Ethernet and Bluetooth devices is 'eui-48'. The instance identifier for each CLA type is the EUI-48 identifier. 'A' has two Ethernet devices, with identifiers '00:1c:bf:93:98:5d' and '00:1b:38:cc:df:ef' and a single Bluetooth device '00:23:6c:9c:a5:f8'. The CLA EIDs of 'A' are as follows:

```
dtn:next:eui-48:00:1c:bf:93:98:5d
```

```
dtn:next:eui-48:00:1b:38:cc:df:ef
```

```
dtn:next:eui-48:00:23:6c:9c:a5:f8
```

4.4.2. The ARs, CLA EIDs and Preferences of 'C'

[TOC](#)

'C' advertises services 'X' and 'Z'. Service 'X' is a type of storage service, that is provides data storage to users of it. Service 'Z' allows execution of commands and is provided to DTN nodes that are connected to an Ethernet segment.

*Service 'X' is accessible through all CLA EIDs of 'C'. 'C' has a Bluetooth CLA with CLA EID 'dtn:next:eui-48:00:23:6C:9C:A5:32', Ethernet CLA with CLA EIDs 'dtn:next:eui-48:00:1c:bf:93:98:c1' and Ethernet CLA with CLA EID 'dtn:next:eui-48:00:1b:38:cc:df:b1'. 'C' has a preferred CLA EID list (see [Appendix B \(CLA Parameters\)](#)) for service 'X' in the order as listed.

*Service 'Z' is to be accessed via the Ethernet CLA with CLA EID 'dtn:next:eui-48:00:1b:38:cc:df:b1' of 'C' only.

4.4.3. 'C' Transmits Bundle containing an EDP ADU

[TOC](#)

'C' periodically transmits bundles that contain EDP ADUs with an EID destination 'dtn::EDPv1' and EID source 'dtn::C', every 30 seconds. Both 'A' and 'B' receive a bundle with an EDP ADU in its payload on their EDP registrations. Both receive the same bundle on each of their CLAs. The EDP ADU received by the EDP registration 'dtn::EDPv1' on 'A' is identical to the ADU received by the registration 'dtn::EDPv1' on 'B'. The ADU contains references to offsets of the data dictionary of the primary block of the bundle which it was the payload. Although the fields contain offsets the actual referenced values are depicted below:

AR EID Reference Count = 3

AR list Ref_scheme_1	AR list Ref_ssp_1
dtn:	next:eui-48:00:23:6C:9C:A5:32
dtn:	next:eui-48:00:1c:bf:93:98:c1
dtn:	next:eui-48:00:1b:38:cc:df:b1

Table 2: CLA EIDs of 'C'

CLA EID Reference Count = 3

CLA list Ref_scheme_1	CLA list Ref_ssp_1
dtn:	:C/X
dtn:	:C/Y

dtn:	:C/Z
------	------

Table 3: AR EIDs of 'C'

The DCF field (below) shows that a bundle with an EDP payload was not requested. This means an EDP report is not generated by 'A' or 'B' in response. As a bundle with an EDP payload was not requested the EDP ADU did not contain a 'Delay' field

The DCF field has the following bits set

Bit	Setting
0	0
1	1
2	1
3	0
4	0
5	0
6	0

Table 4: DCF of 'C'

The values in the DCF field indicate the following about 'C':

- *Bundle Security is enabled
- *DTN node is prepared to be a custodian to any neighbouring DTN nodes
- *Reactive fragmentation is enabled
- *A bundle with an EDP payload is not requested
- *A random response request is not requested

The 'SC' and 'SA' fields (below) indicate that the storage capacity of 'C' is 97 GB and its available storage is 14 GB and can calculate that the storage used is '81352328' or about 86%

The storage fields 'SC' and 'SA'

SC	SA
100790004	14317764

Table 5: Storage of 'C'

The preference list of 'C' (below) contains values that are an index into the reference counts of CLA EID and AR EID. A CLA EID count is followed by a delimiter and then an integer. The integer is an indicator of preference of CLA EID(s) for an active registration.

PREF CLA Reference Count = 4

PREF list Ref_AR_I	PREF list Ref_CLA_I
1	1,1
1	2,2
1	3,3
2	3,1

Table 6: Preference List of 'C'

The preference list shows that the service with AR EID indicated by '1' has three CLA EIDs, indicated by '1', '2' and '3', through which it will receive bundles. 'C' would prefer to use CLA EID identified by '1' with preference indicator '1'. The next preference of 'C' is CLA EID identified by '2' with preference indicator '2'. The CLA EID, that is the least preferred, is identified by '3' with preference indicator '3'. The preference list shows that the service with AR EID that is indicated by '2' has one CLA EID indicated by '3', through which it will receive bundles. A default preference indicator of '1' is used. The table belows show the values that are indicated by the values in the preference list of the EDP ADU of 'C'

PREF CLA Reference Count = 4

PREF list Ref_AR_I	PREF list Ref_CLA_I
dtn::C/X	dtn:next:eui-48:00:23:6C:9C:A5:32
dtn::C/X	dtn:next:eui-48:00:1c:bf:93:98:c1
dtn::C/X	dtn:next:eui-48:00:1b:38:cc:df:b1
dtn::C/Y	dtn:next:eui-48:00:1b:38:cc:df:b1

agent of 'C' reassembles the fragmented bundle and the service with active registration 'dtn::C/X', receives and processes the ADU.

4.4.5. 'B' Uses Service of 'C'

[TOC](#)

An application running on 'B' desires to execute a command on 'C' which is offering a command execution service, identified by endpoint 'dtn::C/Y'. 'B' can determine Bundle Security is disabled on 'C' from the DCF field.

In order for 'B' to transmit the command to 'dtn::C/Y', 'B' generates a bundle with the command execution as data as its payload and transmit it via a DTN next hop of 'dtn:next:eui-48:00:1b:38:cc:df:b1'. The source EID of the bundle is 'dtn::B/Y', destination EID is 'dtn::C/Y', and the next hop is via the Ethernet identifier '00:1b:38:cc:df:b1'. 'B' has two Ethernet devices. Only one ('00:1c:bf:9c:98:5f') is on the same segment as 00:1b:38:cc:df:b1, and has CLA EID of 'dtn:next:eui-48:00:1c:bf:93:98:5d' and this will be used to communicate with 'C'.



Figure 4: DTN Next Hop for service dtn::C/Y

Bundles with EID destination 'dtn::C/Y' and source 'dtn::B/Y' are received by 'C' via its Ethernet identifier '00:1b:38:cc:df:b1', with CLA EID 'dtn:next:eui-48:00:1b:38:cc:df:b1'. The BP agent of 'C' receives the bundle and delivers the ADU to the service with active registration 'dtn::C/Y'.

5. Acknowledgements

[TOC](#)

The DTNRG

[TOC](#)

6. IANA Considerations

This memo includes no request to IANA.

7. Security Considerations

[TOC](#)

A bundle with an EDP payload raises no security considerations beyond those addressed in [\[RFC5050\]](#) ([Scott, K. and S. Burleigh, "Bundle Protocol Specification," November 2007.](#)).

Attacks built on the "Request EDP" operation could exploit the possible side effects of evaluating selection expressions.

DoS-mitigation in DTNs is still a research area. However, it is recommended that bundles are authenticated [\[BPsec\]](#) ([Symington, S., Farrell, S., Weiss, H., and P. Lovell, "Bundle Security Protocol Specification," February 2010.](#)) as a way of making the bad actor accountable.

8. References

[TOC](#)

8.1. Normative References

[TOC](#)

[BPsec]	Symington, S., Farrell, S., Weiss, H., and P. Lovell, "Bundle Security Protocol Specification," draft-irtf-dtnrg-bundle-security-15.txt, work-in-progress, February 2010.
[RFC2119]	Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels," BCP 14, RFC 2119, March 1997 (TXT, HTML, XML).
[RFC5050]	Scott, K. and S. Burleigh, "Bundle Protocol Specification," RFC 5050, November 2007 (TXT).
[dtnURI]	Fall, K., Burleigh, S., Doria, A., and J. Ott, "The DTN URI Scheme," draft-irtf-dtnrg-dtn-uri-scheme-00.txt, work-in-progress, September 2009.

8.2. Informative References

[TOC](#)

[RFC3986]	Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax," STD 66, RFC 3986, January 2005 (TXT, HTML, XML).
-----------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------

[RFC4838]	Cerf, V., Burleigh, S., Hooke, A., Torgerson, L., Durst, R., Scott, K., Fall, K., and H. Weiss, " Delay-Tolerant Networking Architecture ," RFC 4838, April 2007 (TXT).
-----------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Appendix A. DTN Routing Protocols And Algorithms

[TOC](#)

DTN routing protocols and algorithms have EID registrations as they are services used by DTN applications and the BPA. For examples of the the application of EIDs to DTN routing protocols and algorithms please see [\[dtnURI\] \(Fall, K., Burleigh, S., Doria, A., and J. Ott, "The DTN URI Scheme," September 2009.\)](#).

An EDP agent must be able to report a list of the preferred routing protocol or algorithms for each active CLA EID registration. It is expected that DTN routing protocols and algorithms (to be agreed upon), make use of information learned via EDP and provide statistics in such a way that they are accessible to an EDP agent.

Appendix B. CLA Parameters

[TOC](#)

An EDP agent must be able to report a list of the preferred CLAs for each EID registration. An EDP node is expected to be capable of obtaining statistics that include the contact attempts, up-time, number of contacts and throughput of available CLAs. The number of bundles deferred and number of bytes transmitted, queued, and cancelled are also expected to be available to the EDP agent.

CLA parameters that are common to all CLAs SHOULD be available for an EDP node to query. Examples of such common parameters are Link name, CLA EID, CLA type, CLA state, next DTN hop, and MTU.

It is expected that there are CLA parameters that are specific to a CLA. CLA specific data such as the local COMM port, MAC address, IP address, port, whether segment ack or negative ack is enabled and the segment length should also be accessible to EDP

It is expected that an API (see [Appendix C \(APIs for EDP\)](#)) will be required to facilitate the query of an EDP nodes CLA names and parameters.

Appendix C. APIs for EDP

[TOC](#)

It is expected that various APIs are required to enable EDP to query and set parameters. APIs are required to facilitate manipulation of the data dictionary, query CLA names and parameters, query active EID

registration URIs and registration parameters, and to set the DTN hop limit to 1.

Appendix D. The Data Dictionary

[TOC](#)

The EDP application agent must be able to add and delete EIDs to the data dictionary of the primary block (see [Appendix C \(APIs for EDP\)](#)). EDP discharges this responsibility by directing the node's BP application agent to construct the record and manipulate the data dictionary. The data dictionary is populated with a byte array, expressed as a SDNV, of the concatenation of any number of null-terminated active EIDs and CLA EIDs.

Appendix E. Discovery Mechanisms

[TOC](#)

An EDP agent is expected to discover and report active CLAs. EDP is expected to leverage existing discovery schemes when available. At least one CLA, whether discovered and activated automatically or configured and activated manually, is required by EDP.

Authors' Addresses

[TOC](#)

	Alex Mc Mahon
	Trinity College Dublin
	Distributed Systems Group
	Department of Computer Science
	Trinity College
	Dublin 2
	Ireland
Phone:	+353-1-896-2354
Email:	alex.mcmahon@cs.tcd.ie
	Kevin Fall
	Intel Labs, Berkeley
	2150 Shattuck Avenue, #1300
	Berkeley, California 94704
	USA
Phone:	+1-510-495-3014
Email:	kfall@intel.com