Network Working Group
Internet-Draft
Intended status: Informational
Expires: December 22, 2013
A. Detti
S. Salsano
N. Blefari-Melazzi
Univ. of Rome "Tor Vergata"
June 20, 2013

IP protocol suite extensions to support CONET Information Centric Networking
draft-detti-conet-ip-option-05

Abstract

The Information Centric Networking (ICN) paradigm shifts the focus of networking from providing connections between hosts to efficiently providing content to the users. The work on ICN has traditionally been performed looking at "clean-slate" solutions which aims to replace IP with a new paradigm. On the other hand, in this memo we propose an "integration" approach to Information Centric Networking, i.e. we extend the IP protocol suite by defining a new IP Protocol type (CONET). Then we propose two ways of carring ICN related information in IP packets, one uses a new IP Option (both for IPv4 and IPv6), the other one only relies on the IP payload. The ICN related information is used by network nodes and end nodes to support networking based on content rather than (or better in addition to) end-point addresses.

Table of Contents

1. Introduction

In this memo we propose two variants of a solution to support Information Centric Networking [Koponen07][Jacobson09] in IP networks. The proposed solution needs a new Internet Protocol Number to identify the ICN protocol (that we call CONET). The first variant

of the solution is based on extending the IP protocol by using a new IP Option called CONET IP option (defined both for IPv4 [RFC0791] and IPv6 [RFC2460]). The CONET IP option can be used by routers to support content aware networking, in addition to classical address based networking. The CONET IP option is used to identify the content which is the object of the data transfer. Its usage allows efficient in-network caching and replication of content. This solution has been described in [CONET11] and was the only one proposed in previous versions of this draft, up to version 04. The second variant of the solution carries all ICN related information within the IP payload, with no need to extend the IP protocol itself.

The CONET reference architecture foresees End-Nodes, Serving Nodes and CONET nodes (see Figure 1). End-Nodes request for content. Serving Nodes provide content. CONET nodes: i) forward content requests from End-Nodes to Serving Nodes; ii) deliver content from Serving Nodes to End-Nodes; iii) may cache content and therefore provide it to End-Nodes without contacting the Serving Node. CONET nodes can be further classified in Border Nodes and Internal nodes. Border Nodes are able to perform both "forward-by-name" and caching, Internal nodes are not able to perform "forward-by-name" (but only plain IP routing) and can only perform caching.

```
                 requests for content
                 ------------------->
                 content is provided
                 <-------------------
     +----+                            +----+      +----+
     |    |                          --|    |------|    |
     +----+\                        /  +----+      +----+
            \    +----+      +----+ /
             ----|    |------|    |/
                 +----+      +----+
   End-Node      legacy    Intermediate   Border     Serving
                 IP router     Node        Node        Node
       |                                   |
       +---------CONET next hop----------->+
     |     CONET Sub System (CCS) x        |    CCS y     |
```

Figure 1: CONET architecture

As shown in Figure 1, the CONET Information Centric Network can be seen as the inteconnection of CONET Sub Systems (CSSs). A CSS contains CONET nodes and exploits an under-CONET technology to transfer data among CONET nodes. A CSS could be: i) a couple of nodes interconnected by a point-to-point link, e.g. a PPP link or a UDP/IP overlay link; ii) a layer-2 network, e.g. Ethernet; iii) a

layer-3 network, e.g. a private/public IPv4 or IPv6 network, or a whole IP Autonomous System, or even the whole current Internet.

2. ICN information header

In this section we present the information header that contains the ICN related information. In the first variant of our solution, this information will be carried in the IP options, while in the second variant it will be included at the beginning of the IP payload.

The ICN information header has the following format and content:

```
+--------+--------+--------+--------+
|pppLLSCr|  DS&T  |     ICN-ID      |
+--------+--------+--------+--------+
|  optional ICN-ID continuation     |
|    (variable length)  ...         |
+--------+--------+--------+--------+
|CSN(opt)|optional CSN cont.   ...  |
+--------+--------+--------+--------+
| optional extensions (TLV fields)  |
+--------+--------+--------+--------+
```

Figure 2: ICN information

ppp : CONET Information Unit Type - This three bits field is used to differentiate between different types of CONET Information Units (CIUs)

```
0     Reserved
1     Interest CONET Information Unit (Interest CIU)
2     Named-data CONET Information Unit (Named-data CIU)
3-7  Reserved
```

LL : ICN-ID Length Specification - This two bits field provides the length of ICN Identifier (ICN-ID) field or specifies how the ICN-ID length is provided.

0 16 bytes length
1 Reserved
2 ICN-ID field starts with a one byte length field (ICN-ID length in bytes)
3 Reserved

S : Sequence number indication - This one bit field tells if a chunk Sequence Number fiels is present in the Option after the ICN-ID field

0 No Chunk Sequence Number field is present
1 Chunk Sequence Number field is present after the ICN-ID field

C : cache indication - This one bit field is used to control cache operations.

0 No cache
1 Cache

Within Information Units that request for content (e.g. interest CIU), if the bit is set to "No cache" it indicates to the crossed nodes not to look for the content in the cache, but to forward the request toward the source. Within Information Units that carry content (e.g. named-data CIU), if the bit is set to "No cache" it indicates to the crossed nodes not to cache the content.

r : reserved - This one bit field in the first byte after the option length is reserved.

DS&T : Diffserv and Type - This one byte field is used to differentiate quality of services that can be provided by the network to the delivered content and to identify the content type. This field can be used to encode the content type and the priority as follows:

```
+--------+
|Fxxxxxxx|
+--------+

+--------+
|0TTTTPPP|
+--------+

+--------+
```

```
                    |1TTTTTTT|
                    +--------+
```

The righmost bit can be considere as a flag F. If the flag bit F is set to 0 the three rightmost bits encode 8 priority levels and other 4 bits are for the content-type. If the flag bit is set to one, no preallocated semantic to the remaining bits is given.

ICN-ID : ICN Identifier (ICN-ID) field - The ICN-ID is a unique identifier for the content. The ICN-ID is carried in the ICN-ID field. How to determine the length of this field is defined by the ICN-ID Length Specification field. If the ICN-ID Length Specification field determines the field length, the ICN-ID field only carries the ICN-ID. If the ICN-ID Length Specification field indicates that the field length is carried in the field itself, the ICN-ID field starts with a one byte field that determines its length.

If ICN-ID Length Specification = 0 (i.e. 16 bytes len), the ICN-ID field is as follows:

```
                +--------+--------+--------+--------+
                |               ICN-ID              |
                +--------+--------+--------+--------+
                |                                   |
                +--------+--------+--------+--------+
                |                                   |
                +--------+--------+--------+--------+
                |                                   |
                +--------+--------+--------+--------+
```

If ICN-ID Length Specification = 2 (i.e. ICN-ID starts with a one byte length field), the ICN-ID field is as follows:

```
                +--------+--------+--------+--------+
                | length |       ICN-ID               |
                +--------+--------+--------+--------+
                |                ...                |
                +--------+--------+--------+--------+
                |                ...                |
```

The ICN-ID starts with a two bytes field called ICN-ID namespace ID that determines the structure of the rest of the ICN-ID. ICN-ID namespace values needs to be assigned by the IANA. Note that in most circumstances, the ICN-ID can be processed by the routers as an

opaque object, as described in Section 4. This is why the ICN-ID namespace ID has been included at the beginning of the ICN-ID itself. In other cases the nodes are requested to perform a forward-by-name procedure, which may require a semantic understanding of the ICN-ID.

```
             +--------+--------+--------+--------+
             | ICN-ID namesp ID|       ...       |
             +--------+--------+--------+--------+
             |                ...                |
             +--------+--------+--------+--------+
             |                ...                |
             +--------+--------+--------+--------+
             |                ...                |
             +--------+--------+--------+--------+
```

CSN : Chunk Sequence Number - This optional field carries the Chunk Sequence Number that identifies a portion of the content. When a content is split in a sequence of smaller unit called "chunks", this field can explitly carry the sequence number of the chunk (another solution is obvioulsy to embed the chunk number in the ICN-ID). The Chunk Sequence Number is represented with a variable number of bytes. An initial bit pattern determines the length of the CSN field.

```
1 byte CSN (7 bits CSN range)
    +--------+
    |0       |
    +--------+

2 bytes CSN (14 bit CSN range)
    +--------+--------+
    |10               |
    +--------+--------+

3 bytes CSN (21 bit CSN range)
    +--------+--------+--------+
    |110     |        |        |
    +--------+--------+--------+

4 bytes CSN (28 bit CSN range)
    +--------+--------+--------+--------+
    |1110    |        |        |        |
    +--------+--------+--------+--------+

5 bytes CSN (32 bit CSN range)
    +--------+--------+--------+--------+
```

```
       |11110000|        |        |        |
       +--------+--------+--------+--------+
       |        |
       +--------+
```

6 bytes CSN (40 bit CSN range)

```
       +--------+--------+--------+--------+
       |11110001|        |        |        |
       +--------+--------+--------+--------+
       |        |        |
       +--------+--------+
```

Binary patterns from 11110010 to 11111111 are reserved. They can be used to extend the CSN range if needed. With the above defined option, we can have up to 2^40 chunks in a content. Assuming a relatively small chunk size of 1 KBytes, it is possible to have a content of 1099 TeraBytes, while assuming a more reasonable chunk size of 256 Kbyte it is possible to have a content of 281474 TeraBytes (218 PetaBytes).

The rationale for having a variable length encoding is the following. The CSN range for a given content is determined by the content size divided by the chunk size. As content of very different sizes can be transmitted, the CSN range can be very different. Therefore it is not efficient to dimension this field considering the maximum number of chunks in which a content can be split.

3. CONET protocol

A specific IP protocol number needs to be assigned to the CONET protocol:

CONET IP protocol number : xxx (to be assigned by IANA).

The figure below shows the CONET protocol stack. CONET protocol is divided in two sub-layers, whose data unit are respectively denoted as "Carrier Packets" and "CONET Information Units". A CONET Information Unit (CIU) can be split into different Carrier Packets. Each Carrier Packet is transported by an IP packet. There are different types of CONET Information Units, the CIU type information is carried in the CONET Information Unit Type field in the CONET IP option.

```
       +--------+--------+--------+ \
       | CONET Information Units  |  |
```

```
     +--------+-------+--------+  |
                                  |
     +--------+-------+--------+  |- CONET protocol
     |     Carrier Packets     |  |
     +--------+-------+--------+  |
                                  |
     +--------+-------+--------+ /
     | IP (opt. CONET IP option)|
     +--------+-------+--------+
```

Figure 3: CONET protocol layers

The generic structure of a Carrier Packet (CP) is reported hereafter:

```
     +------------------------+
     |    CP Payload header   |
     +------------------------+
     |       CP Payload       |
     +------------------------+
     |      CP Path state     |
     +------------------------+
```

The ICN information described in the previous section can be transported either in the IP Option or at the beginning of the CP Payload header. The CP Payload header includes specific information for each CIU type and can depend on the "transport" protocol. It will be described in other specification documents. The definition of a receiver driven ICN transport protocol called ICTP (Information Centric Transport Protocol) is proposed in [I-D.ICTP] (see also [ICTP12]). The CP payload header contains the length of the CP Payload and allows to identify the start of the CP Path state field. The CP Path state field can be used in End-Nodes, Border Nodes and Serving Nodes to assist in the forwarding operation of carrier packet, therefore it is described here.

The CP Path State field stores the End-Node address and the addresses of the set of crossed Border Nodes in the path from End-Node to the Serving Node (or to a border or Intermediate Node that provides a requested content). The format of the CP path state field is reported hereafter (assuming that IPv4 addresses are carried). The use of CP Path state is optional, as the path from End-Node to Serving Node can be stored in the so called PIT (Pending Interest Table) in the crossed nodes [Jacobson09]. A node crossed by an Interest packet can either add its address to the CP Path State (and create the CP Path State field if not present) or store the pending interest in the PIT.

CP Path State field

```
+--------+--------+--------+--------+
|0  len  | pointer| ad-type| addr 1 |
+--------+--------+--------+--------+
| addr 2 | addr 3 | addr 4 | ad-type|
+--------+--------+--------+--------+
| addr 1 | addr 2 | addr 3 | addr 4 |
+--------+--------+--------+--------+
|                ...                |
+--------+--------+--------+--------+

+--------+--------+--------+--------+
|1      len       |     pointer     |
+--------+--------+--------+--------+
| ad-type| addr 1 | addr 2 | addr 3 |
+--------+--------+--------+--------+
| addr 4 | ad-type| addr 1 | addr 2 |
+--------+--------+--------+--------+
| addr 3 | addr 4 |
+--------+--------+
```

The length field specifies the length of the CP Path State field in bytes. If the first bit of the len field is 0, the remaining 7 bits of the first byte are used as len field and both the length field and the pointer field are one byte length. In this case the maximum value of the length of the CP Path State field is 127. If the first bit of the len field is 1, both the length field and the pointer field are two bytes length. In this case the maximum value of the length of the CP Path State field is 32767.

The pointer field specifies the offset, starting from the start of the CP Path State field where the last address has been inserted.

Each address is represented as a couple (ad-type, address) it could be represented by a triple (ad-type, ad-length, address) if the address type is of variable length. The ad-type field is one byte size and currently admitted values are:

```
0     Reserved
1     Public IPv4 address (len is 4 bytes, no ad-length needed)
2     Public Ipv6 address (len is 16 bytes, no ad-length needed))
3     Ethernet address (len is 6 bytes, no ad-length needed))
4-255 Reserved
```

4. Procedures

4.1. Interest CONET Information Unit (Interest CIU)

4.1.1. Processing in the End-Node

An end-node that wants to retrieve a content (or better a Chunk of a content) issues an Interest CIU, the ICN-ID and the Chunk Sequence Number of the required Content are respectively transported in the ICN Identifier (ICN-ID) field and in the CSN field of the ICN information header. The end-node stores its IP address in CP path state field, initializing the pointer field. Assuming for simplicity that the Interest CIU will fit into a single Carrier Packet, the Interest CIU will be included in the Carrier Packet that in turn is inserted into an IP packet.

The end-node must now determine the destination IP address for the Carrier Packet. The end-node performs a forward-by-name operation, trying to associate the ICN-ID with a next hop (i.e. with the IP address of the next hop). The next hop can be the Serving Node (if the Serving Node is in the same CONET Sub System of the end-node) or a Border Node of the CONET Sub System (if the Serving Node is in a different CONET Sub System). Typically the End-Node does not participate to the content routing protocols, therefore it cannot resolve the ICN-ID into the address of the next hop, but it has to ask an external entity, behaving in a similar way of a current name server (such external entity could be a part of a system that handles the content routing, called Routing Name System). Once this information is retrieved, the end-note can fill the IP destination address in the IP header and sends the packet. The end-node may cache the mapping (ICN-ID -> next hop) into its memory as well.

4.1.2. Processing in the Serving Node

If the Serving Node is in the same CONET than the end-node, the Serving Node IP address will be used a destination IP address by the end-node. The Serving Node will receive an IP packet directed to itself, whose IP protocol number is "CONET". Therefore the packet will be internally dispatched toward the "CONET entity" in the Serving Node. The CONET entity reads the CONET information unit type from the CONET IP options and recognizes that the received packet is an interest packet. Then it reads the ICN-ID and Chunk Sequence Number in the ICN information header, the ICN-ID will correspond to a content provided by the Serving Node. The CONET entity will then process the CONET transport protocol information carried in the IP payload, which may for example specify a requested offset within the chunk. Finally the CONET entity will respond to the interest packet by sending the requested named-data CIU.

4.1.3. Processing in the Border Node

If the Serving Node is in a different CONET Sub System than the end-node, the address of a CONET Border Node will be used a destination IP address by the end-node. The Border Node will receive an IP packet directed to itself, whose IP protocol number is "CONET". Therefore the packet will be internally dispatched toward the "CONET entity" in the Border Node. The CONET entity reads the CONET information unit type from the ICN information header and recognizes that the received packet is an interest packet. Then it reads the ICN-ID and Chunk Sequence Number in the ICN information header and is able to understand which content and which part of the available content it needs to provide. If the Cache indication field is set to "No Cache" or if the field is set to "Cache" but the chunk is not available in the cache, the Border Node starts the forward-by-name process. It will resolve the next hop of the interest packet, which can be a Serving Node in a different CONET Sub System (with respect to the one from which the interest packet was received) connected to the Border Node, or another Border Node in the path toward the Serving Node. Before sending out the packet, the Border Node adds its IP address in the CP Path State field and updates the pointer field. Note that these procedures needs to be performed in the "fast path" of the Border Node (in this case the CONET entity in the Border Node can be seen as an integral part of the enhanced IP protocol). If the Cache indication field is set to "Cache" and the Border Node has found that the chunk corresponding to the ICN-ID/CSN is available in its cache, the Border Node will process the CONET transport protocol information carried in the IP payload, which may for example specify a requested offset within the chunk and it will respond to the interest packet by sending the requested named-data CIU.

4.1.4. Processing in the Intermediate Node

When a packet is sent to the CONET next hop (as selected by the End-Node or by a Border Node) using the IP destination address of the next hop resolved by the forward-by-name, it can cross different IP routers in the path from the sending node and the next hop. A crossed router that is aware of the ICN information header, is a CONET Intermediate Node. This node may have cached the the chunk that is requested by the interest packet. The Intermediate Node works as follows. When processing the IP header for the received packet, it finds that the packet contains the CONET IP protocol. If the Cache indication field is set to "No Cache", the Intermediate Node forwards the packet using the destination IP address. If the Cache indication field is set to "Cache", the Intermediate Node checks the presence of the chunk in its cache before forwarding the IP packet. Therefore, it reads the ICN-ID and Chunk Sequence Number in the ICN information header and checks if the chunk is present in

its cache. If the chunk is not present, the normal IP processing is continued. Note that these operations needs to be performed in the "fast path" of the router and they only require information that is transported in the IP option. If the chunk is present in the CONET router cache, the router will process the CONET transport protocol information carried in the IP payload, which may for example specify a requested offset within the chunk and it will respond to the interest packet by sending the requested named-data CIU.

4.1.5. Processing in the legacy routers

When a packet is sent to the CONET next hop (as selected by the End-Node or by a Border Node) using the IP destination address of the next hop resolved by the forward-by-name, it can cross different IP routers in the path from the sending node and the next hop. If a crossed router is a legacy router not aware of the CONET protocol, it will simply forward the packet looking at the IP destination address. If the ICN information header is carried in the IP Option, a requirement for such legacy router is to be configured not to drop IP packets carrying unidentified IP options.

4.2. Named data CONET Information Unit (Named data CIU)

4.2.1. Processing in the responding node

The responding node is the node that is able to provide a content (identified by ICN-ID and Chunk Sequence Number) to a requesting end-node. Therefore the responding node can be a Serving Node which provides an original copy of the content, or a Border Node / Intermediate Node that provide a cached copy of the content. The responding node will use the Path State information contained in the received carrier packet carrying the Interest CIU to forward back the carrier packets containing the named-data CIU towards the requesting end-node. In particular, it will use the pointer field to read the last address in the list and will use it as IP destination address for the Carrier packet carrying the named-data CIU. We can denote this address as "CONET previous hop". Then it will update the pointer field so that the next node will use the previous address in the list. It may choose to strip the used address from the list in the CP Path state, thereby reducing the CP Path State field length.

4.2.2. Processing in a Border Node

The Border Node will receive an IP packet directed to itself, whose IP protocol number is "CONET". Therefore the packet will be internally dispatched toward the "CONET entity" in the Border Node. The CONET entity reads the CONET information unit type from the ICN information header and recognizes that the received packet is a

named-data packet. Again, we stress that this processing should be performed in the fast path. Being a named-data packet, the Border Node will read the CP Path State field in the Carrier Packet and by using the pointer field will identify the CONET previous hop in the path towards the requesting end-node. Before sending out the packet, it will update the pointer field in the CP Path State field. The destination IP address of the packet will be set to the CONET previous hop retrieved from the CP Path State field. If the Cache indication bit in the IP option is set to "Cache", the Border Node may choose to cache the CIU that is transported by the carried packet. In this case, it is reccomended that the Border Node dispatches the packet as soon as possible and operates on a local copy to perform cache related operations.

4.2.3. Processing in an Intermediate Node

An Intermediate Node, i.e. a router in the path between a Serving Node or a Border Node and the CONET previous hop, which is aware of the CONET option, may decide to cache the named data CIU transported by a carrier packet. The Intermediate Node will receive an IP packet with an IP destination equal to the CONET previous hop and will immediately forward this packet using IP routing. Then, if the Cache indication bit in the IP option is set to "Cache", the Intermediate Node may choose to cache the CIU that is transported by the carried packet.

4.2.4. Processing in the legacy routers

When a packet is sent to the CONET previous hop (as selected by the Serving Node or by a Border Node) using the IP destination address of the previous hop obtained using the CP Path State information, it can cross different IP routers in the path from the sending node and the previous hop. If a crossed router is a legacy router not aware of the CONET IP protocol, it will simply forward the packet looking at the IP destination address. If the ICN information header is carried in the IP Option, a requirement for such legacy router is to be configured not to drop IP packets carrying unidentified IP options.

5. Forward-by-name framework

The forward-by-name process is performed in the end-node and in Border Nodes in order to resolve a ICN-ID into the next hop towards a Serving Node for the given ICN-ID. This document provides a framework under which the forward-by-name procedures can be performed, and assures that different forward-by-name procedures and approaches may coexist. These different approaches needs to be separately specified. The format and the semantic of the ICN-ID may need to be specified when defining a specific forward-by-name

approach. This is made possible by the concept of ICN-ID name space ID, which is carried within the ICN-ID.

The basic procedure that a forward-by-name framework needs to offer is called resolveICN-ID, it takes as input the ICN-ID and returns the next_hop_address. This procedure is performed by end-nodes and by Border Nodes that are not able to provide a cached response for a content requested by an End-Node.

resolveICN-ID (ICN-ID) -> next_hop_address

The tables on which the forward-by-name procedures are based are populated by Serving Nodes and by Border Nodes. The procedure is initiated by Serving Nodes that advertize the hosted content with the advertizeICN-ID procedure. In turn, the procedure is replicated by the Border Nodes that spread the received advertising toward other Border Nodes. This procedure takes as input a ICN-ID, the address of the node performing the procedure, and the path information towards the Serving Node as seen by the node performing the procedure. Depending on the specific content routing approach, the path information can be simply an hop count, or it could be the path list (as in the BGP AS-PATH).

advertizeICN-ID (ICN-ID, node_address, path_info)

In the following section we define two CONET default name spaces. It could be more appropriate that in future version of this document this specification is provided in a separate document.

6. CONET default namespaces

We define two default ICN-ID name spaces for CONET, one is based on variable length strings as ICN-ID, as it was proposed in [Jacobson09], the second one is based on fixed length hashes. The two namespaces are assigned the following ICN-ID name space IDs.

Namespace ID	
1	VLL (Variable Length Label) ICN-ID namespace
2	PLHB (Principal/Label Hash Based) ICN-ID namesp.

In the VLL (Variable Length Label) CONET namespace the ICN-ID is simply the string representation of a resource. As described in [Jacobson09] ICN-IDs are hierarchically structured so that an individual name is composed of a number of components (see [Jacobson09] for further details. An authority is needed to ensure the uniqueness of the ICN-IDs. The approach should be similar on how the uniqueness of DNS names is granted in today's Internet.

In the Principal/Label Hash Based CONET namespace the ICN-ID is the composition of two hash values, as follows:

ICN-ID = (hash (Principal) , hash (Label))

In the Principal/Label Hash Based CONET namespace the Hash(principal) is a 8 bytes hash of a string representing the Principal. The Label is a 6 bytes hash of a string representing the label. A central authority is needed to ensure the uniqueness of the Hash(principal), i.e. a Principal cannot be assigned if its hash collides with an already assigned hash. The Principal is responsible to ensuring that each Hash(Label) belonging to the Principal are unique. Therefore a Label cannot be used by a Principal if its hash collides with the Hash of an already used Label.

7. Two ways of carrying ICN information in IP packets

Two ways of carrying the ICN information in IP packets have been considered. The first way introduces new IP Options to be included in IPv4 and IPv6 headers, the second option simply includes the ICN information header at the beginning of the CP payload header, that is within the IP payload. Hereafter, the details related to the definition and use of the IP Options are given, then the two ways are compared.

7.1. Using CONET IP Option to carry ICN information

The CONET IPv4 option has the following format:

```
+--------+--------+--------+--------+
|100xxxxx|yyyyyyyy|pppLLSCr|  DS&T  |
+--------+--------+--------+--------+
|      ICN-ID (variable length)     |
|                 ...               |
+--------+--------+--------+--------+
|CSN(opt)|optional CSN cont.   ...  |
+--------+--------+--------+--------+
| optional extensions (TLV fields)  |
+--------+--------+--------+--------+
```

Figure 4: CONET IP Option for IPv4

The CONET IPv6 option has the following format:

```
+--------+--------+--------+--------+
|001xxxxx|yyyyyyyy|pppLLSCr|  DS&T  |
+--------+--------+--------+--------+
|      ICN-ID (variable length)     |
|                ...                |
+--------+--------+--------+--------+
|CSN(opt)|optional CSN cont.   ...  |
+--------+--------+--------+--------+
| optional extensions (TLV fields)  |
+--------+--------+--------+--------+
```

Figure 5: CONET IP Option for IPv6

For IPv4 the first byte (the option type) is as follows:

```
Type:
  Copied flag:  1 (all fragments must carry the option)
  Option class: 0 (control)
  Option number: xxxxx (decimal) TO BE ALLOCATED BY IANA
```

For IPv6 the first byte (the option tyep) is as follows:

```
Type:
  Unrecognized option action : 00
          (skip option, process the rest of the header)
  Change allowed flag        : 0
          (option data cannot change while the datagram is en route)
  Option number: xxxxx (decimal) TO BE ALLOCATED BY IANA

Length:
  yyyyyyyy: variable length of IP option in bytes (including the
            Type and Length bytes
```

7.2. IPv6 handling of CONET option

The IPv6 CONET option has to be interpreted by all routers in the path that are ICN capable. Therefore we it naturally fits into the the IPv6 Hop-by-hop header, which is the first extension header that can be present after the fixed part of the header. The Hop-by-hop header is meant to be read by all routers in the path.

7.3. Comparison among the two ways

If the IP Option is used, a CONET ICN packet can be identified by the presence of the IP option. Otherwise, it has to be identified by looking at the IP Protocol type information in the IP header.

The first advantage of the solution based on IP options is conceptual: it allows an ICN Node to take routing decision by considering the information contained in the layer 3 (IP) header. Moreover, is a packet gets fragmented at IP level, each fragment keeps the IP Option in its IP header, allowing the processing of the single fragments at ICN level. The disadvantages are: legacy IP nodes could have some problems with unrecognized IP options (experiencing higher processing times or even dropping such packets); a more complex implementation in end nodes, as it requires changes in the IP layer. In [CONET11] we investigated (with practical experiments on PlanetLab) how our unrecognized IP option is handled by current routers in the Internet. In the large majority of tests it was possible to add unrecognized options and achieve end-to-end CONET connectivity among arbitrary PlanetLab nodes, while in few cases some routers in the path have dropped the packets. IP Options have often been criticized because their support in current routers would impose a performance penalty, but may be we can assume here that routers will be modified to support Information Centric Networking so this performance issue may not be critical.

The advantages of the other approach (not using the IP options) are complementary: the implementation in terminals is simpler and legacy IP nodes are not affected by the ICN information carried in the IP payload. A disadvantage of this solution is that when IP packets gets fragmented, the IP fragments loose the ICN information header. Strictly speaking, a node operating in this approach is not just a router, as it is using layer 4 information to process the packets so it can be seen as a middlebox [RFC3234] capable of performing ICN routing (and caching) functionality. This disadvantage is not critical in nodes that are in any case capable of operating with transport layer information (e.g. a node with an SDN/OpenFlow architecture [ICN-SDN13]).

8. IANA Considerations

This document requires the allocation of one IP protocol number by the IANA.

This document requires the allocation of one IP option by the IANA if the solution of using IP options is adopted

This document requires that IANA will maintain the registry of CONET namespaces.

9. Security Considerations

Security considerations to be provided

10. References

10.1. Normative References

[RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, September 1981.

[RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.

10.2. Informative References

[CONET11] A. Detti, et al., ., "CONET: A Content Centric Inter-Networking Architecture", ACM SIGCOMM Workshop on Information-Centric Networking (ICN-2011), Toronto, Canada , August 2011.

[I-D.ICTP] Salsano, S., Detti, A., Blefari-Melazzi, N., and M. Cancellieri, "ICTP - Information Centric Transport Protocol for CONET ICN", draft-salsano-ictp-02 (work in progress), June 2012.

[ICN-SDN13] Blefari-Melazzi, N., Detti, A., Morbito, G., Salsano, S., and L. Veltri, "Information Centric Networking over SDN and OpenFlow: Architectural Aspects and Experiments on the OFELIA Testbed", conditionally accepted (minor revisions) to Elsevier Computer Networks special issue on Information-Centric Networking - arXiv preprint arXiv:1301.5933 , 2013.

[ICTP12] S. Salsano, et al., ., "Transport-layer issues in Information Centric Networks", ACM SIGCOMM Workshop on Information-Centric Networking (ICN-2012), Helsinki, Finland , August 2012.

[Jacobson09] V. Jacobson, et al., ., "Networking named content", Proc. of ACM CoNEXT 2009 , 2009.

[Koponen07]
T. Koponen et al., ., "A data-oriented (and beyond) network architecture", Proc. of ACM SIGCOMM 2007 , 2007.

[RFC3234] Carpenter, B. and S. Brim, "Middleboxes: Taxonomy and Issues", RFC 3234, February 2002.

Appendix A. Acknowledgments

We acknowledge the financial support by the EU in the context of the CONVERGENCE and OFELIA research projects

Appendix B. Document history

draft-detti-conet-ip-option-05

- o new title "IP protocol suite extensions to support CONET Information Centric Networking"
- o added the generic ICN information header and a second variant of the solution not using the IP Option

draft-detti-conet-ip-option-03

- o new title "IPv4 and IPv6 Options to support Information Centric Networking"
- o added IPv6 support with IPv6 Option

Authors' Addresses

Andrea Detti
Univ. of Rome "Tor Vergata"
Via del Politecnico, 1
Rome 00133
Italy

Email: andrea.detti@uniroma2.it

Stefano Salsano
Univ. of Rome "Tor Vergata"
Via del Politecnico, 1
Rome 00133
Italy

Email: stefano.salsano@uniroma2.it

Nicola Blefari-Melazzi
Univ. of Rome "Tor Vergata"
Via del Politecnico, 1
Rome 00133
Italy

Email: blefari@uniroma2.it