ICNRG
Internet-Draft
Intended status: Informational
Expires: July 17, 2021

I. Moiseenko
Apple Computer
D. Oran
Network Systems Research and Design

January 13, 2021

Flow Classification in Information Centric Networking
draft-moiseenko-icnrg-flowclass-07

Abstract

For the ubiquitous and highly important Internet protocols (TCP, UDP, IP), flows are conventionally identified by the "5-tuple" of source and destination IP addresses, source and destination port, and protocol type in an IP packet. Information Centric Networking (ICN) is a new paradigm where network communications are accomplished by requesting named content, instead of sending packets to destination addresses. This document describes mechanisms allowing ICN forwarders, consumers, producers and other ICN nodes to encode, decode, and process equivalence class identifiers (flows) at any desired granularity of a routable name prefix and beyond the routable name prefix. This document is a product of the IRTF Information-Centric Networking Research Group (ICNRG).

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

The problem of identifying groups of packets that get consistent treatment in a network and allowing that treatment to be independent and isolated from the treatment of other groups of packets, is ubiquitous and long-standing. The purposes to which this identification can be put is highly varied, including such functions are providing differentiated quality of service, traffic engineering, traffic filtering for security functions like intrusion detection and firewalling, etc.

Providing the capability to apply different functions to groupings (formally equivalence classes) of packets is generally known as the "flow identification problem" where the definition of what constitutes a "flow" is highly dependent on the particular protocol or protocols carrying the packets. Some of the above uses of flows also bring a mechanism requirement that the flow identification technique be useful to have not just equivalence classes, but the ability to apply some useful notion of fairness among the instances of each equivalence class. There are many possible flow identification techniques that are either too granular (spatially or

temporally) to establish fairness, or conversely too coarse and cannot separate traffic a fine enough level to have useful fairness.

For the ubiquitous and highly important Internet protocols (TCP, UDP, IP), flows are conventionally identified by the "5-tuple" of source and destination IP addresses, source and destination port, and protocol type in an IP packet. Some systems augment this by further distinguishing equivalence classes by the TOS/DSCP field, but this is secondary to the 5-tuple methods. 2-party flows are present where the source and destination addresses are unicast IP addresses. Multi-party flows can exist when the destination IP address is a multicast address. One key common characteristic is that the identification of flows depends in a very deep way on the presence of source addresses in the packets, and the limited richness of IP addresses is correspondingly constraining as a means to classify traffic in a semantically meaningful way.

The purpose of this document is to devise a mechanism allowing ICN forwarders, consumers, producers and other ICN nodes to encode, decode, and process equivalence class identifiers (flows) at any desired granularity of a routable name prefix and beyond the routable name prefix.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

2. Flow Identification Challenges and Opportunities in ICN

ICN systems differ from IP-based designs in a number of ways, three of which are quite fundamental.

1. The packets are addressed to a rich namespace of packets, which is hierarchical and carry semantic information that can be useful for classification of flows.

2. Conversely, the packets do not contain source addresses of any kind, which means that identifying flows as groups of packets between a single pair of endpoints (in the unicast case) is not possible for intermediate forwarders (other than possibly the first-hop forwarder if it serves a single consumer per interface).

3. Instead of group-based multicast, ICN systems use multi-destination delivery semantics. This allows a different way to map packets to flows, and in fact in the IP world multicast has

been difficult to use partly because there is no good way to make use of flow identification for multicast flows (for a variety of reasons).

These differences lead to a need to find a different method to identify flows than used in the IP protocol suite. Ideally, the method would provide semantics that map well with the expected uses of ICN to build applications. It would also use native capabilities in the ICN protocols rather than having to change the protocol architecture in ways that affect the semantics or utility of an ICN approach to networking.

In NDN and CCN protocols, Interest and Data names are the only identifiers in the network; neither source addresses nor destination addresses are employed. Each Interest packet is responded by exactly one Data packet, producing a useful property known as "flow balance". This means that flow identification can be tied directly to the Interest/Data exchanges. The key to having useful flow identification is for the equivalence classes to be associated with the names in the corresponding Interest and Data packets, and to be stable over multiple exchanges using different names that share some common "handle" that can be used to separate the names into equivalence classes. As mentioned above, simply using the routing state that maps name prefixes to routes does not provide a useful set of equivalence classes, because:

- in general, routing prefixes are too coarse; many equivalence classes of packets are generally covered by a single routing prefix because they are present at the same set of destinations from a routing perspective;

- practical, scalable routing needs to do route aggregation, which further blurs the discrimination of the equivalence classes.

Therefore, NDN and CCN protocols need to have something that both relates to the name structure but provides finer granularity for flow classification purposes. This document describes two alternative mechanisms addressing these issues.

3. Flow Encoding Schemes

Flow encoding schemes described in this document allow ICN systems to perform flow identification at any desired granularity of a routable name prefix and beyond the routable name prefix. Techniques described herein permit both consumer nodes and forwarders to use equivalence classes to perform per-flow functions. The encoding to achieve the flow classification is lightweight and does not require changes to the protocol architecture in ways that affect the

semantics or utility of an ICN approach to networking. Furthermore, equivalence classes can be specified by the data producer, in contrast to IP protocols in which the data producer can only control the destination port as an equivalence-class discriminator.

No matter what method is used to identify equivalence classes that can be treated as flows, there is the independent but critically important issue of how to scale any state that is kept on a per-flow basis when the flow count is very high. For consumers and producers, this state scales naturally with the number of applications and application interactions are going on simultaneously. Therefore the scaling limit is not likely to be in the producers or consumers. For ICN forwarders this state could scale quadratically or worse if the forwarders need classic prior resource reservation to deterministically partition resources on a producer/consumer pair basis. This need not be the case however. Practical resource control algorithms exist that keep state only for "active" flows (those with packets either currently or recently moving through the network). Further state reduction is also possible with some loss of accuracy using approximate techniques, like stochastic fair queuing (SFQ). If the ICN forwarder cannot keep all the state due to memory or processing limitations, it faces the common problem of which flows to remember and which to forget. This problem is fundamental, and is mostly independent of the choice of flow identification method in the protocol. Flow encoding schemes described in this document provide a method for identifying equivalence classes using protocol machinery that already has to scale (e.g. name parsing and lookup) and hence does not introduce a new class of problems not inherently present.

3.1. Equivalence class component count (EC3)

For this encoding scheme a new field called equivalence class component count (EC3) is introduced into the Data packets. It is set by a producer and counts the number of name components in the corresponding name that are to be considered, when grouped together under the same prefix part of the name, to be one equivalence class instance. This allows either finer (or coarser) granularity than provided by routing prefixes. Because the EC3 is a separate field of the packet (Figure 1), producers can "regroup" equivalence classes dynamically by including more or fewer levels of the name hierarchy when they respond to Interests for the corresponding Data packets. This brings a set of clear advantages and disadvantages. The primary advantage is flexibility in re-grouping equivalence classes, especially in aggregating flows at different granularities. The main disadvantage is that the binding of the equivalence class into the namespace is not explicit, and hence it is harder to enforce consistent interpretation among producers, consumers and forwarders.

An additional consideration with the EC3 encoding scheme is whether or not the field is inside or outside the security envelope that provides cryptographic packet integrity to the name and data in the data packet. Either approach is possible; however having the field outside the security envelope would allow ICN forwarders to modify it, allowing the aggregation/disaggregation of flows to be performed by the forwarders as well as the consumers. Conversely, leaving the field outside the security envelope may enhance certain attack scenarios against flow classification when employed for quality of service differentiation or firewall filtering.

```
+---------------------------------------------------------------------+
|  /youtube |  /<mediaID>   |  /video  OR |  <frameID>   | <segment#> |
|           |               |  /audio     |              |            |
+-----------+---------------+-------------+--------------+------------+
| Name      | Name          | Name        | Name         | Segment    |
| component | component     | component   | component    | component  |
| type      | type          | type        | type         | type       |
+-----------+---------------+-------------+--------------+------------+
|                                                                     |
| Equivalence Class Component Count = 2 (up to MediaID stream)        |
|                           OR                                        |
| Equivalence Class Component Count = 3 (video or audio substream)    |
+---------------------------------------------------------------------+
```

An example of EC3 encoding of flow information.

Figure 1

3.2. Equivalence class name component type (ECNCT)

For this scheme the equivalence class information is encoded directly in the name, by adding a name component to the name of the Interest and Data packets. This new typed named component is called equivalence class name component type (ECNCT). It is set by the producer as part of constructing all Data packets in the desired equivalence class and is therefore immutable for the lifetime of the associated named data. A consequence of this is that the ECNCT is present in Interest packets as well, and hence may affect both PIT matching and FIB matching. The Equivalence Class name component both names the equivalence class explicitly, and implicitly makes all Data packets named below it in the hierarchy part of that equivalence class. In other words, the name can have multiple equivalence class (e.g. flow and subflows) markings using this scheme (Figure 2). As in EC3 encoding scheme, depending where in the name component hierarchy the ECNCT is placed, one can have either finer or coarser granularity than provided by routing prefixes.

The exact details of how to encode the ECNCT name component may differ among ICN architectures. The CCN design has explicitly typed name components, so for that protocol an explicit name component type can be assigned straightforwardly. The NDN design eschews typed name components and instead uses textual naming conventions for name components. In that case an architectural constant string would be chosen to distinguish ECNCT from other name component semantics.

/youtube	/<mediaID>	/video OR /audio	<frameID>	<segment#>
Name component type	Flow component type	Flow component type	Name component type	Segment component type

An example of ECNCT encoding of flow information.

Figure 2

When an ICN forwarder receives a packet with a name carrying ECNCT(s), it can be processed on a component-by-component basis, and substreams can be identified according to name prefixes indicated by the equivalence class identifiers. The identification of substreams enables special treatment of selected substreams. For example, video substreams can be discriminated from other substreams, such as audio substreams. In the example in Figure 2, two name components include equivalence class identifiers to define a hierarchy of flows (or substreams). Specifically, two flow components are encoded to define the following hierarchy of flows:

First level name prefix: /youtube/<mediaID>

Second level name prefix: /youtube/<mediaID>/video

Second level name prefix: /youtube/<mediaID>/audio

4. Producer operation

In ECNCT encoding scheme, an ICN producer receives an Interest packet carrying equivalence class identifiers in the name. A producer might use the equivalence class identifiers for demultiplexing, load sharding and other purposes, and reply with a Data packet matching the Interest name.

In EC3 encoding scheme, an ICN producer receives an Interest packet that might not carry an equivalence class identifier. In such case,

the producer may refer to the name schemas used in a particular application to dynamically determine the equivalence class identifier for Interest demultiplexing, load sharding and other purposes, and for replying with a Data packet carring the equivalence class identifer in EC3 field.

5. Consumer operation

An ICN consumer may also use the knowledge of equivalence classes of packets to take certain actions. For example, when a Data packet with a name specifying a particular equivalence class arrives at a consumer in response to a previously sent Interest packet, the consumer can associate the data packet with the correct equivalence class. Consequently, the consumer can manage subsequent Interest/Data exchanges with the same name prefix and equivalence class identifier (e.g., EC3 or ECNCT) as one flow. Associated measurements such as round trip time (RTT) or marginal delay can be leveraged to perform flow and congestion management for the equivalence class as a whole.

6. Forwarder operation

A flow table may be provisioned in ICN node to enable the node to make decisions about performing actions on Interest and/or Data packets based on one or more equivalence classes. The flow table can include name prefixes mapped to equivalence class identifiers obtained from previous Interest-Data exchanges. In ECNCT encoding scheme, Interest packets carry the equivalence class identifier, therefore flow table may only include name prefixes. Typically, name prefixes in flow table are more granular than prefixes in the FIB, but less granular than names in the PIT. Flow table could be separate from other elements of ICN node or could be integrated with FIB or PIT.

Flow management logic can be configured to treat flows having the same equivalence class similarly. Actions taken that are related to flows or objects having a similar equivalence class can include, but are not limited to, dropping a packet, using a particular interface for a packet, security related actions (e.g., filtering traffic for security functions like intrusion detection and firewalling), quality of service (QoS) related actions (e.g., types of resources to allocate to the packets, moving a packet up in the queue for forwarding purposes, etc.), and/or traffic engineering (e.g., selecting one path over another path). Flow management logic can enable such actions to be taken on a particular flow based on the equivalence class associated with the flow or object and policies related to the equivalence class.

Specific examples of how ICN node can use the knowledge of equivalence classes of packets include, but are not limited to, the following:

1. Enforce rate control for the equivalence class as a whole (e.g., dropping packets, queuing packets, etc.);

2. Estimate the number of simultaneous flows traversing a bottleneck link, which can improve the performance of many congestion control schemes; and

3. Make more intelligent selections of which packets to cache at the ICN forwarder, for example, to prefer to cache many packets of the same equivalence class.

7. IANA Considerations

This memo includes no request to IANA.

8. Security Considerations

Certain attack scenarios against flow classification for quality of service or firewall filtering may be prevented if the EC3 field located inside the security envelope. ICN forwarders can read, but not change, the EC3 value, because the EC3 field is covered by a security signature and not encrypted.

If the EC3 field is outside of the security envelope, it can be placed in the hop-by-hop headers and, therefore, be modified by the transit ICN forwarders. This allows the transit ICN forwarders to override the flow definitions set by the producer applications, but opens the system to various attack scenarios.

Modification of equivalence class identifiers in ECNCT encoding scheme effectively modifies the packet name, and therefore, ECNCT does not introduce any additional security threats.

9. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

Authors' Addresses

Ilya Moiseenko
Apple Computer
USA

Email: imoiseenko@apple.com

Dave Oran
Network Systems Research and Design
4 Shady Hill Square
Cambridge, MA 02138
USA

Email: daveoran@orandom.net