ICN Research Group
Internet-Draft
Intended status: Informational
Expires: September 22, 2016

R. Ravindran
A. Chakraborti
A. Azgin
Huawei Technologies

March 21, 2016

Forwarding-Label support in CCN Protocol
draft-ravi-ccn-forwarding-label-02

Abstract

The objective of this proposal is to enable ID and Locator namespace split in the CCN protocol that has several applications such as towards Interest routing optimization, mobility, handling indirections in manifests, and routing scalability. We enable this through the notion of forwarding-label (FL) object, which is an optional hop-by-hop payload in the Interest message with a locator name which identifies a network domain, router, or a host. Depending on the application and trust context, an FL object can be subjected to policy based actions by the forwarders such as invoking security verification or enabling other service-centric actions such as FL object replacement. FL object can be inserted by the applications or by the network. To enable dynamic name resolution FL objects can also be modified in the network by designated points such as the edge routers.

Table of Contents

1. ID-Locator Namespace Split in CCN

In the context of ICN/CCN, we define identifier and locator as follows:

- Identifier (ID) is a persistent secure or non-secure flat-ID or a hierarchical name assigned to a content, device or service. If the ID is secure, then trust relationship can be derived from it. Generally the identifier space is managed by applications.

- Locator (LID) is a routeable topological name assigned to a network entity such as a router, a server, or an end device. Generally the locator space is managed and assigned by the network administrators.

We discuss here the motivations behind the need for separation between persistent name (ID) and a locator (LID) in the Interest message in the context of CCN and a proposal to achieve this. The advantages of ID/Locator have been extensively studied and it has been part of many host-centric protocols such as HIP[2], ILNP [3], and LISP [4] and is also part of FIA architectures such as MobilityFirst[16]. Specifically in CCN, ID based routing is not efficient considering the need to dynamically replicate content and handle mobile entities, and the problem to address routing scalability [11]. Hence providing this distinct ID-LID separation in the protocol offers the following advantages:

- Today CCN applications bind to persistent IDs, while their resolution is handled through per-hop name-based routing by a CCN forwarder using unicast/anycast/broadcast mechanisms, with routing scalability handled through its namespace aggregation. This model can introduce problems when the named entity is mobile, migrated, or replicated, as the names have to be announced in the routing control plane which can in turn introduce routing instability and scalability challenges (as the name aggregation to topological binding cannot be satisfied anymore). Enabling ID/Locator split and managing this mapping in a separate name resolution service shall address the routing churn introduced by dynamic entities. CCN is unique in the sense that both name-based routing and resolution service can operate simultaneously driven by their use based on a given context. For e.g. while inter-domain routing can be handled using a name resolution service, intra-domain routing can be based on name-based routing.

- ID and Locator namespaces are managed by different entities. IDs are managed by applications, hence relevant only to consumers, producers and intermediate service points, while locator names are managed by a network administrator. Locators map to network domains or specific network elements through which the named entity is reachable. The relationship between the two is established during the namespace publishing phase, and managed by a separate name resolution service. ID/Locator distinction in CCN allows applications to manage its own namespace and not be restricted by the naming rules imposed by the network.

- Affording ID/Locator split in an Interest message offers many advantages in CCN especially when a centralized control is applied such as using NFV/SDN frameworks, enabling efficiency and

flexibility in name resolution, routing, mobility, service chaining, and routing scalability.

Considering the above requirements, we propose a Forwarding-label (FL) object, which acts as a locator and provides the flexibility to forward Interests on a name other than the one provided within the original Interest message with the ability to modify it within the network. Handling ID/Locator mapping requires a control plane infrastructure and appropriate network layer state with security functions to prevent malicious usage. Specific control plane or security mechanism of ID/Locator mapping is out of the scope of this document as many techniques can be used towards achieving this. This draft presents various considerations towards FL object management such as: FL object insertion/modification/deletion, FL object processing by a CCN forwarder, PIT/CS implications for FL object carrying Interests, FL object Interest packet format, and security/ trust considerations. We then discuss the application of FL object in various scenarios.

2. Forwarding-Label Object Proposal

The use of FL is required when routing by ID is inefficient in scenarios such as replicated content, device mobility, or scalability challenges when ID based routing is employed. FL objects are subjected to processing and modification in the network depending on the specific use case scenario. Following we discuss various aspects of FL related to its semantics and management.

2.1. FL Object Naming

FL objects are container objects that include LID, service specific metadata, and security attributes for authentication. LIDs are hierarchically structured topologically names where the names follow the definition in [1]. The security attributes are optional and may include validation payload and algorithm as discussed in [1].

2.2. FL Object Insertion

An FL object can be inserted in an Interest message by the consuming application or by the network.

In certain situations, the application logic may use an FL object in addition to the ID in the Interest message or this action may also be triggered because of feedback from the network, for instance due to failure of routing the Interest message based on the ID. In such situations, forwarders which process traffic from applications outside the trust domain require a way to validate the FL object. A possible approach to ensure trust in such situations is discussed in

[11] where a trust binding is provided between the ID and the LID as a link object which can be validated by the forwarder. To avoid the possibility of a misuse of an FL object, a default policy of the network may be to ignore it from untrusted applications and only choose to route by the content ID.

In the case where the FL object is inserted by the network, FL object insertion is triggered at the ingress service routers of the network domain. For instance, network may insert an FL object to an incoming Interest message, if the Interest message satisfies the flow service profiles that are imposed by the network administrator at the ingress edge routers. The service profile matching actions may include matching an Interest name to a set of service prefixes or triggered by certain markings such as context-ID (for e.g. contexts may include service, trust, location) in the Interest message. FL objects inserted within the trust domain may not require security validation.

In situations where a forwarder handles both of these scenarios, policies can be applied at the ingress router to handle the two cases accordingly. These policies may include the face on which the Interests arrives on, or the Interest IDs etc.

2.3. FL Object Swapping

An FL object can be swapped by another within the network in the context of a given service at designated points, such as the service edge routers, in the network. As an FL object carries a LID, and with appropriate representation and security considerations in the Interest message, FL objects also can be potentially stacked if the Interest message has to be tunneled through a domain, where routing based on the top level FL object is not feasible.

2.4. FL Object Termination

FL objects are terminated by a forwarder when the LID in it matches its own LID. Here we assume a forwarder to possibly have many LIDs such as domain-IDs or router-IDs. For e.g. a forwarder (in a domain) identified as /att/santaclara can process an FL object with its LID set to this router's domain name or to a forwarder ID such as /att/santaclara/pop-x. Whenever an FL object is terminated by the forwarder, depending on the service context, it can attach a new FL object, or conduct additional processing (e.g. re-resolution of the name to a new FL object) based on the Interest parameters. The FL object can also include optional policy metadata based on which FL objects can be swapped in the network.

3. FL Object Message Format

As FL objects are swappable in the network, it is proposed as a hop-by-hop field in the optional body of the fixed header as shown in Figure 1. The optional FL container includes attribute of type FL-Object, which contains a name TLV identifying the LID (Figure 2). LID is a hierarchically structured variable length name as defined in [1]. A LID implies a locator such as an AS-ID, Gateway-ID, Router-ID or Host-ID. In addition to the LID, optional FL metadata includes contextual information on the application or the service to aid the network for invoking an appropriate FL processing, such as trust validation of the FL object. Optional security attributes, such as authentication information, can be included depending on the specific use case scenarios, such as secure name delegation information as discussed in [11], or signature of the consumer.

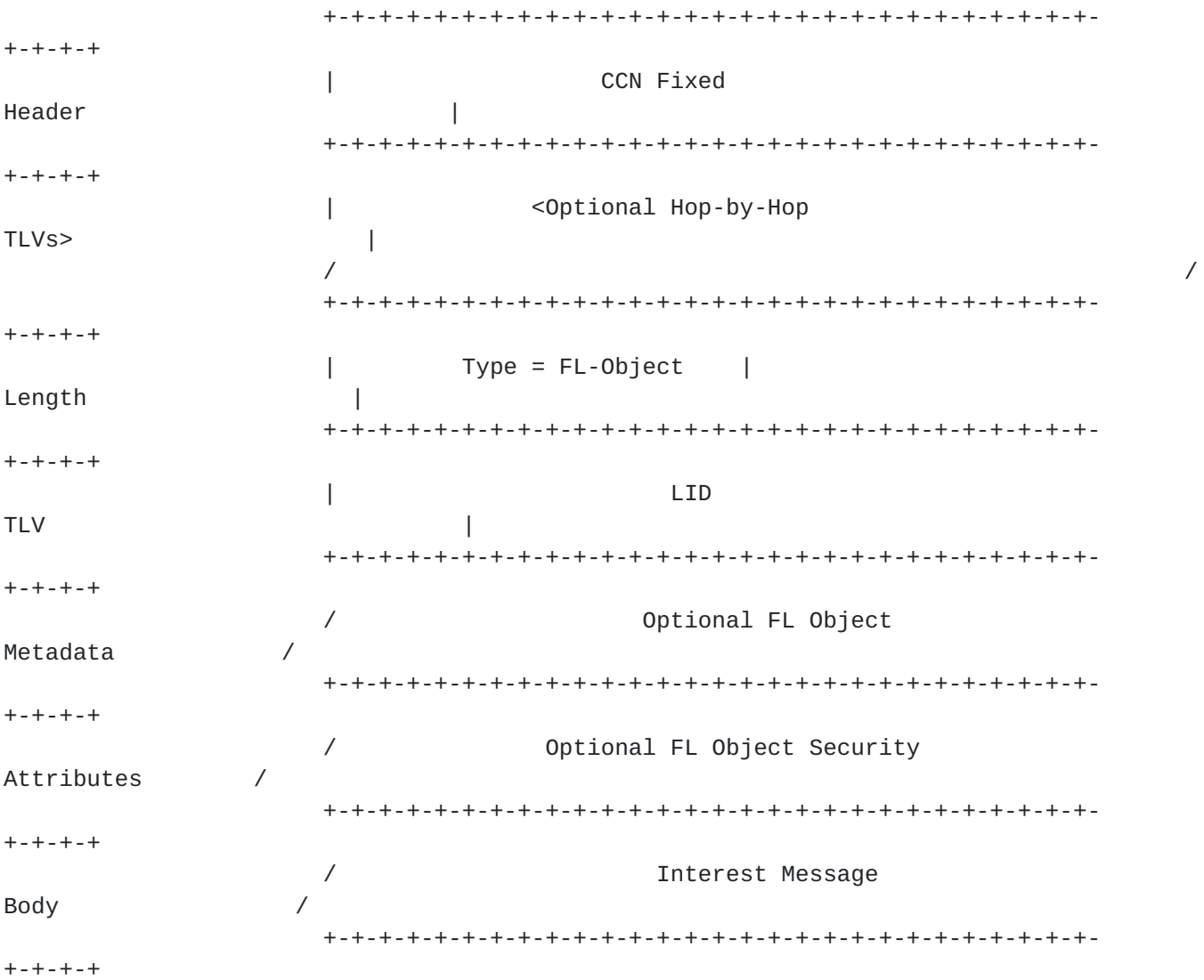

Figure 1: Interest message with hop-by-hop Optional Forwarding-Label TLV

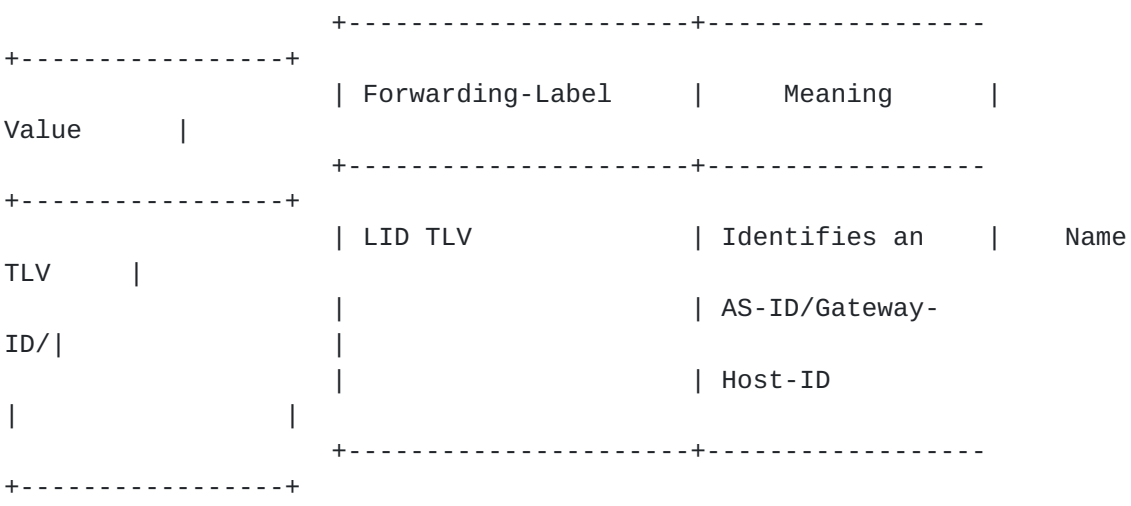

Figure 2: Locator-ID (LID) Definition

4. FL Object Processing Rules

The following discussion is based on the assumption that all forwarders must process the optional header fields. In the context of CCN packet processing, FL object is only relevant when the decision to forward the Interest message is to be made. At this stage, multiple options exist, assuming consistent policies exists throughout the domain: 1) In the first scenario, the rule may be that if an FL object is included in an Interest message, then it should be given preference over the ID. This is under the assumption that FL objects are trusted indirections within the Interest message, and can be validated by the router if required; 2) In the second scenario, the forwarder could prioritize forwarding on the ID, and then forward on the LID at every hop; 3) In the third scenario, where policy based routing is involved, more complex routing approaches can be considered at the network edge, such as the forwarder could apply service policy on the Interest ID and choose to remove or swap it with a new FL object irrespective of the current FL object inserted in the Interest message, while the core nodes could use a more simpler approach, such as approach 1 or 2. Following are the steps when approach 1 is applied.

- During Interest packet processing, while a forwarding decision is being made, if an LID is available then it should be preferred for forwarding over the name in the Interest message irrespective of the feasibility of ID based routing.

- The validation of FL depends on the trust context. In trusted scenarios, where the applications and the network are managed by

the same authority, the forwarder can bypass the validation. In untrusted scenarios, the edge router may validate the FL that is inserted by the sender, and to avoid re-validations by successive forwarders, these Interests can be marked to have been validated at the ingress point.

- If FL object is trusted and validated and the lookup based on LID in the FL object succeeds, then two possibilities exist: 1) for a non-terminating flow, the LID FIB lookup results in a next hop towards which the Interest is forwarded ; 2) for a terminating flow, LID lookup invokes a service logic wherein the service either re-resolves the Interest ID to another LID resulting in a new FL object or removes the current FL object and subjects the Interest to regular processing based on the ID in the Interest message.

- If the FL object is trusted and validated and the LID lookup fails, then the router can try to forward the Interest based on the Interest ID. However if routing based on the Interest ID fails, then the router could raise an error condition and feedback the message to the previous hop, in the same or a different domain, with the appropriate error code.

5. PIT Processing Implications

To maintain the simplicity of the forwarding logic, the purpose of an FL object should be to guide the Interest towards the closest source of the resource entity, hence FL object may only be used for the forwarding decision and not be required for content object processing. However there may be usage scenarios where the FL object state is required to be saved in the PIT and even piggybacked in the content object (CO).

For example, in the case when there is no binding between the ID and the LID in the Interest expressed by a consumer, and multiple Interests arrive carrying the same ID but with different LIDs, then the expected outcome is to forward all such Interests with unique LIDs. In this case the forwarder is required to save the LID along with the Interest ID in the PIT and forward the duplicate Interest whose LIDs differ from ID to LIDs state saved in the PIT.

In another application, it may be required to decouple the choice of one consumer's LID from another consumer's LID, i.e, when a secure binding exists between the ID and the LID. In this case, the forwarder stores the FL object in the PIT, and the returning CO should piggyback the Interest's FL object as long as the CO is from the location intended in the LID, which is matched against the pending PIT entry before continuing with the reverse path forwarding. In cases, where the FL object is swapped by the intermediate routers, the CO should be updated with the appropriate FL to ensure matching of the PIT entries along the previous hops. These considerations are similar to those elaborated in [14].

6. Caching Implications

The considerations here follow our previous discussion, where the FL object is piggybacked in the CO. If there is an implicit security binding between the Interest ID and the LID, then the FL object state is piggybacked along with the CO, and the FL object in the incoming Interest should be matched against the CO's FL object before the cached content object is returned.

7. Multiple Domain Scenario

In wide area network scenarios, Interests cross multiple domains. If an FL object is only trusted within the domain boundaries, then the FL object is removed before the Interest is forwarded to the next domain, which then, upon entry inserts a new forwarding label with the associated security attributes at the ingress of the next domain. But if trust exists between domains, such as one through a trusted third party (validated based on the FL object security binding), to use the FL inserted by the previous domain, then the intermediate domains can avoid further FL processing and use the FL object passed on by the previous domains.

8. FL Object Security

FL object security is related to the purpose it is used for and the control plane mechanism used to manage it. Depending on the use case scenario of the FL, appropriate security mechanisms should be applied to secure the control and data planes to avoid exploitation of this feature.

Generally, the major threats against the FL object approach is to manipulate the relationship between the name and the FL object. Such manipulations can happen in various scenarios, some of which are listed as follows: (i) a malicious interceptor (acting as a publisher) intentionally injecting an incorrect mapping into the name resolution system; (ii) a malicious interceptor (between the edge router and the resolution server) manipulating the mapping sent back from the name resolution system when the edge router queries the mapping system ; (iii) a compromised intermediate router maliciously changing the FL object, e.g., with the wrong FL object or an out-dated FL object; (iv) an untrusted application injecting invalid FL object into the Interest message.

To achieve network level FL security, appropriate mechanisms should be applied to provide mapping provenance, mapping integrity and to prevent replay attack to address these issues. The security mechanisms applicable to the above discussed scenarios (i) and (ii) are similar to ones applied to secure other mapping systems such as

LISP [5] and DNS[7]. Scenario (iii) requires new security mechanisms, one such way is to enable a domain level trust infrastructure so that the mapping between the name and the FL object can be authenticated by the successive routers.

In untrusted environments, when an FL object is inserted in the Interest message by the end hosts, appropriate authentication information should also be included in the FL object to allow ingress routers to optionally validate the delegation of the Interest ID to LID [11]. Furthermore, additional security policies can be enabled by the network to handle FL objects outside its trust domain.

9. Use Case Scenarios

Here we provide the discussions related to using FL objects in different scenarios.

9.1. Handling Producer Mobility

In the literature the different techniques to handle producer mobility can be classified into the following two types:

- Application-based approach, where the application takes the responsibility for announcing its reacheability to the network and triggering a network state change to enable Interest routing towards the mobile producer. Most of the current proposals fall under this category, and these include the following two approaches: 1)The Kite proposal [13] implements an anchor based approach where consumers and producers agree on an anchor point based on external mechanisms and uses application initiated (traced and tracing) Interests to handle the producer mobility; 2)The anchor-less proposal [15] is another application-based approach wherein enhanced name-based routing is used to track the mobile producer. While these approaches allow consumers and producers to work on a single name space, it raises scalability concerns with increasing number of mobile nodes and number of applications signaling into the network. As these approaches introduce more signaling in the network, the operational efficiency of packet forwarding is negatively affected due to the state changes that have to be applied to maintain the sanity of the Interest packet processing logic. Another potential security issue with these approaches is that it can be prone to flooding attacks by malicious applications targeting specific application names and impeding their normal operation.

- Network-based approach uses the late-binding technique [8], wherein the reachability of the mobile node is handled by the network. In this case applications can explicitly request

mobility for a given name space [9], with the network handling the mobility by tracking its latest location in the network through a name resolution system. At the same time, through coordination of the old and new point-of-attachments (PoA) and in-band signaling one can achieve zero loss for a given Interest flow. The late-binding technique uses ID/Locator split that is only applied at the PoAs, thereby avoiding any routing churn in the network due to producer mobility, while offering better scalability when the number of mobile producers increases. Here the mobile entity (ME) registers a persistent name that requires mobility with its current point-of-attachment (PoA). The PoA then registers the mapping between the name and the PoA's locator in its local name resolution system. Then the domain updates the ME's home domain name resolution system with its current domain LID. When a correspondent nodes expresses Interest for the name, it is first resolved to the current ME domain by the home domain. When the Interest enters the domain offering mobility service, it is resolved again to the ME's current location. Furthermore PoA-to-PoA signaling can be enabled to offer seamless forwarding of Interests whenever an ME changes its PoA. In addition to correcting the path stretch the Interests re-routed from the old PoA can be marked and re-routed to the new PoA with the new FL. On the return path, the CO are also marked, this in-band marking is used by the ingress PoA at the consumer's end to re-resolve the mobile prefix to a new forwarding locater that would correct the path stretch.

9.2. Manifests

The FL objects can also be used to support the retrieval of nameless objects [10]. Using the current manifest proposal [6] a consumer receives a manifest with the ContentObjectHashIDs and their respective locator information. A consuming application uses the locater as a routeable content name, while the ContentObjectHashID is used as a HashID restriction parameter. Multiplexing the Interest name field as an ID and also as an LID has the following consequences: (1) a forwarder cannot distinguish between Interest packets containing ID or LID in the name field, as the protocol doesn't differentiate these two constructs; (2) it complicates Interest processing when LID is used as a name, by first requiring to check for the presence of ContentObjectHashID, and to use it to index the Interest based on it instead of the locator name; (3) more complications arise if an Interest packet arrives with two IDs i.e. a ContentObjectHashID as the hash restriction and the ID as the content name, in which case, one of them may seek precedence over the other.

The above issues can be avoided through the use of the ContentObjectHashID as the content name and the locater in the FL

object. In this case, a forwarder will always index the pending Interest table based on the content name. The routing decision then would be based on the FL object depending on the routing policy in the forwarder. This also avoids the situation of dealing with two IDs in the Interest packet, i.e. the application has to choose either ID or ContentObjectHashID as the content ID. This use of FL object can be enforced in a straight forward manner by identifying flat-ID, e.g. ContentObjectHashID, and routeable name as different typed name objects in the Interest packet.

A possible high level forwarding logic for the edge/core router to support nameless objects based on the above discussion is presented in Figure 3. Here edge router can also be a gateway node.

```
  Begin

  if Edge_Router
     If Interest arrives on a face with a flat-ID
        Then check for the presence of FL object
          If FL object is present, use the LID in the FL object for Interest
forwarding

        If there no FL Object
          If policy allows, resolve the flat-ID with a NRS to obtain an FL
object
            Use the FL object to route the Interest

     End

    If the Interest arrives with a routeable ID
      If there is FL object
        Then use the ID for forwarding and Remove the FL object

      If there is no FL object
        Match Interest ID with name policy for e.g. mobility or interest
routing optimization
          If a name policy for resolution exists
            Then network resolution service is invoked on the ID  which returns
an FL object
             Use the FL object to direct the Interest to the appropriate next
hop
    End
  End

  if Core_Router
      if Interest arrives with an FL Object
        Use the LID for forwarding
      Else if Interest is with a Routeable ID
        Use the name for forwarding
  End
```

Figure 3: Forwarding logic to support flat-ID and routeable ID at the edge router

We discuss security implications of using ID and FL object in the Interest message depending on the ID type and

- Case 1 - ContentObjectHashId with FL object: This use case is a straight forward simplification of what is being proposed in [10].

Here the locator is included within the FL object and the ContentObjectHashId is used as the name, so this shouldn't introduce any new security concerns. This holds good for both application and network based FL object insertion.

- Case 2 - Routeable ID with FL object: For UE based FL object insertion, this scenario can cause cache poisoning in the absence of signature check enforcement in the forwarders. For the case when FL object is managed within a trusted domain, the security implications are discussed in Section 8.

9.3. Interest Routing Optimization

Networks which hosts its own or third party content/service can benefit from the ability to handle Interest routing logic in its domain opportunistically. When a Interest seeking a specific content or service enters a network domain, the ingress router can redirect the Interest to the closest cache point or service location.

9.4. Routing Scalability

As discussed in [11], locator based routing can address routing scalability as the number of ASs are many orders less than the number of information objects. This reduces the forwarding table in the DFZ zone in the order of number of ASs in the Internet.

10. Informative References

[1] CCN Wire format, CCNX1., "http://www.ietf.org/id/draft-mosko-icnrg-ccnxmessages-00.txt.", 2013.

[2] Nikander, P., Gurtov, A., and T. Henderson, "Host identity protocol (HIP): Connectivity, mobility, multi-homing, security, and privacy over IPv4 and IPv6 networks", IEEE Communications Surveys and Tutorials, pp: 186-204, 2010.

[3] Atkinson, R., "An Overview of the Identifier-Locator Network Protocol (ILNP)", Technical Report, University College London, 2005.

[4] LISP, RFC6380., "https://tools.ietf.org/html/draft-ietf-lisp-sec-07.", 2014.

[5] LISP-Security, LISP-SEC., "https://tools.ietf.org/html/draft-ietf-lisp-sec-07.", 2014.

[6] CCNx, Manifest., "http://www.ccnx.org/pubs/draft-wood-icnrg-ccnxmanifests-00.html.", 2015.

[7] DNS-SEC, RFC4033., "DNS Security Introduction and Requirements.", 2005.

[8] Afanasyev, A., "Map-and-Encap for Scaling NDN Routing.", NDN Technical Report ndn-004-02, 2015.

[9] Ravidran, R., "Realizing Mobility as a Service in CCN.", IETF/ICNRG, Paris Interim 2016, 2016.

[10] Mosko, M., "Nameless Objects.", IETF/ICNRG, Paris Interim 2016, 2016.

[11] Azgin, A., Ravindran, R., and G. Wang, "A Scalable Mobility-Centric Architecture for Named Data Networking.", ICCCN (Scene Workshop) , 2014.

[12] Cisco System Inc., CISCO., "Cisco visual networking index: Global mobile data traffic forecast update.", 2009-2014.

[13] Zhang, Y., Zhang, H., and L. Zhang, "Kite: A Mobility Support Scheme for NDN.", NDN, Technical Report NDN-0020 , 2014.

[14] CCNx Label Forwarding, CCNLF., "http://www.ccnx.org/pubs/ccnx-mosko-labelforwarding-01.txt.", 2013.

[15] Auge, J., Carofiglio, G., Grassi, G., Muscariello, L., Pau, G., and X. Zeng, "Anchor-less Producer Mobility in ICN.", ICN, Sigcomm, 2015 , 2015.

[16] NSF FIA project, MobilityFirst., "http://www.nets-fia.net/", 2010.

Authors' Addresses

Ravishankar Ravindran
Huawei Technologies
2330 Central Expressway
Santa Clara, CA 95050
USA

Email: ravi.ravindran@huawei.com

Asit Chakraborti
Huawei Technologies
2330 Central Expressway
Santa Clara, CA 95050
USA

Email: asit.chakraborti@huawei.com

Aytac Azgin
Huawei Technologies
2330 Central Expressway
Santa Clara, CA 95050
USA

Email: aytac.azgin@huawei.com