

ICNRRG
Internet Draft
Intended status: Informational
Expires: January 2015

J. Hong
ETRI
W. Chun
Hufs
H. Jun

g

ET

RI

July 20, 201

4

Bloom Filter-based Flat Name Resolution System for ICN
draft-hong-icnrg-bloomfilterbased-name-resolution-00.txt

Status of this Memo

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

This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

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/lid-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 January 21, 2015.

Copyright Notice

Copyright (c) 0000 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Abstract

In information-centric networking (ICN), uniquely identifiable and location independent names are assigned directly to the named data which raises scalability issues and they get even worse with flat names. Accordingly, name resolution system required for lookup-by-name routing in ICN has to be designed to scale, also considering mobility support. In this draft, a bloom filter-based flat name resolution system (B-NRS) is proposed where the bloom filter as an aggregated form of names and hierarchical structure of the B-NRS are exploited to address the scalability issues.

Table of Contents

1. Introduction	3
2. Bloom filter-based name resolution system (B-NRS)	4
2.1. System structure.....	4
2.2. Key operations	5
2.2.1. Name registration.....	5
2.2.2. Locator Update.....	5
2.2.3. Lookup	6
3. Performance analysis.....	6
4. Security Considerations.....	6
5. IANA Considerations	7
6. Conclusions	7
7. References	7
7.1. Normative References.....	7
7.2. Informative References.....	7
A.1. Authors' Addresses.....	9

1. Introduction

In contrast to the host-centric networking in the current Internet, the primary communication object in information-centric networking (ICN) is named data, where uniquely identifiable and location independent name is assigned directly to the named data. This shift raises scalability issues to a new level. The current Internet is addressing on the order of 10^9 nodes, whereas the number of addressable ICN objects is expected to be several orders of magnitude higher [ICNRG charter]. Accordingly, name resolution system required both for lookup-by-name routing in ICN [ICN Challenges] and for ICN-IoT architecture [ICN-IoT] has to be designed to scale, also considering mobility support.

In this draft, we propose a bloom filter-based flat name resolution system (B-NRS) which maintains and resolves the binding between names and locators, i.e. B-NRS takes a name as its input and produces the locator sets that the name is currently associated with. We assume that the locator independent names are flat since the flat names provide some advantages compared to hierarchical ones, such as higher flexibility, simpler name allocation and benefits in terms of persistency and privacy [Ghodsi, ITU]. On the other hand, scalability becomes the most important challenge on designing the NRS supporting flat names. It is because of the ever increasing number of names in the network and no possible way to compactly represent the flat names such as the aggregation in IP addresses.

In order to address the scalability issue in designing the NRS for flat name, we need to aggregate names in any shape of type. One popular technique for flat name is Distributed Hashing Table (DHT) based approach [Hanka, Luo, Ahlgren, Mathy], where multiple servers form circular linked list and the bindings are stored in the appropriate server. However, the DHT technique has some drawbacks; the binding must be stored in a server other than the owner's, which causes a serious trust problem related to the authority issue and lookup message may be propagated through the long paths.

In this draft, to overcome the drawbacks of DHT, we exploit the bloom filter as an aggregated form of names and hierarchically construct the B-NRS. One of the major benefits of the bloom filter is a fixed constant time of insertion and search which is completely independent of the number of names already in the set. Another important and powerful property of bloom filter is the efficient support for union of bloom filters with the same size and set of hash functions which can be implemented with bitwise OR. However, bloom filter also has some drawbacks; false positive and no member deletion. Although there is no way to get rid of the false positive,

it can be minimized by choosing the right parameters. The deletion problem is also taken care by periodic reconstruct of the bloom filters or by using variants of the bloom filter such as the counting bloom filter.

We note that the B-NRS in this draft does not require any specific mechanism for registering names, since names have no structure and can be registered to any B-NRS server with no constraint. Thus, the B-NRS needs only lookup mechanism. Whereas in the DHT-based system, the lookup message for a name is forwarded by the same way how to register the name.

2. Bloom filter-based flat name resolution system (B-NRS)

We propose a bloom filter-based name resolution system (B-NRS) for supporting flat name which maintains and resolves the binding between names and locators.

2.1. System structure

We construct the B-NRS hierarchically by defining a network of B-NRS servers, which consists of a forest by several disjoint trees. The network of B-NRS servers is defined by both parent-child and peering relationships.

A B-NRS server consists of a name lookup table which stores the binding between names and locators for all names which are directly registered to the BRS server. The lookup table takes an name as the input and produces its associated locator sets as the output.

We utilize bloom filters as an aggregated form of names at each B-NRS server. B-NRS servers announce their name set to the other B-NRS servers. Instead of announcing the whole list of names, bloom filter as an aggregated form of names is announced. When announcing its name set to its peers or parents, the B-NRS server announces the union of name sets of all child B-NRS servers. Union of child name sets can be built by using the characteristic of bloom filter that bloom filter for union of sets can be built merely by bitwise 'OR' operation on all the sets. Thus, each B-NRS server stores bloom filters for itself, from children, and from peers.

We note that the forest of B-NRS servers retains the loop-free property for the use of bloom filter.

At the top of the trees, the B-NRS servers are fully peered, which means that each server shares its knowledge of all names that it manages with its the peers. A leaf B-NRS server knows every single

name/locator pair that it manages but nothing else. The intermediate B-NRS servers know the name/locator pair for all names that are directly registered to them and also possess only information about the names that their descendant and peer B-NRS servers manage.

2.2. Key operations

2.2.1. Name registration

When a communication entity attempts to join the network, it must register itself in at least one B-NRS server. In this draft, it is allowed that the communication entity can be registered in any arbitrary B-NRS server since names have no structure.

Upon receiving the registration request from the communication entity, the B-NRS server registers the name to its lookup table. The locators for the name are stored in the table when the communication entity for the name is actually present into the network. We separate this as the operation of locator update from the name registration.

The name registration is along with bloom filter update. When a communication entity is registered in a B-NRS server, the registration information is extracted from its name using the hash functions for its bloom filter and inserted into its own bloom filter first and then the B-NRS server updates bloom filters for its parents and peers, where this recursion holds until bloom filters at the top of trees are completely updated.

When names are deleted from the lookup table, we need to adopt a certain mechanism to update the bloom filters for the deletion since bloom filter cannot handle the deletion by itself. Thus, we use the periodic refresh technique that bloom filters with registered names are rebuilt periodically and followed by bloom filter updates.

2.2.2. Locator Update

When a communication entity actually presents in the network, the locator update is occurred, where the gateway sends the locator update message to the correspondent B-NRS server and the locator associated with the name is stored in the lookup table. If the name has multiple locators, then they are stored as a set of locators for the name. Through the bloom filter test of the name, the locator update messages are forwarded into the lookup table where the name is stored.

When the communication entity depresents from the network, the locators for the name is deleted from the lookup table by the locator update message as well. Thus, changing locators has no effect on the structure of the B-NRS and mobility is easily supported.

2.2.3. Lookup

The lookup operation is to find the locator information for a given name. The simplest case is when the source object tries to communicate with the destination object registered in the same B-NRS server. B-NRS server always searches for the destination name in its own lookup table first so the locator information is acquired at the first lookup in such a case.

A harder, but more interesting, case is when the destination object is registered in the other B-NRS server with the source object. In this case, the B-NRS server would quickly learn that the destination object is not registered in the same B-NRS server by a simple search of its lookup table. Then, it searches bloom filters for its child and peer B-NRS servers. If none of the bloom filters return a positive answer, the lookup request message is forwarded to its parent B-NRS server. On the other hand, if any of bloom filters return a positive answer, the lookup request message is forwarded to every B-NRS server that corresponds to the bloom filters with positive answers. We note that because of the false positives of the bloom filter, multiple bloom filters may return positive answers.

This search is done recursively, and the locator information for the destination name can eventually be found. Once the locator information is found, it is delivered to the source object by the lookup reply message which takes the reverse path of the lookup request message.

3. Performance analysis

TBD

4. Security Considerations

TBD

5. IANA Considerations

TBD

6. Conclusions

In this draft, we proposed a bloom filter-based name resolution system (B-NRS) supporting flat name. The proposed system is a network of B-NRS server in a forest by multiple trees with peering relationship. Scalability issue was addressed by information compression using bloom filters to represent collection of names of the child B-NRS server, where the bloom filter was exploited to overcome some drawbacks of DHT-based system. The peering relationship was adopted to alleviate the traffic load to the B-NRS servers at the upper part of the B-NRS.

7. References

7.1. Normative References

7.2. Informative References

[ICNRG charter] <http://irtf.org/icnrg>

[ICN Challenges] D.Kutscher, S. Eum, K. Pentikousis, I. Psaras, D. Corujo, D. Saucez, T. Schmidt, and M. Waehlich, "ICN Research Challenges ", draft-kutscher-icnrg-challenges-02, February 2014.

[ICN-IoT] Y. Zhang, D. Raychadhuri, R. Ravindran, and G. Wang, "ICN based Architecture for IoT", draft-zhang-iot-icn-architecture-01, June 2014.

[Ghodsi] A. Ghodsi, T. Koponen, J. Rajahalme, P. Sarolahti, and Shenker, "Naming in Content-Oriented Architectures," In Proceedings of the SIGCOMM ICN'11, August 19, 2011, Toronto, Ontario, Canada.

[ITU] International Telecommunication Union (ITU), "ITU-T Recommendation Y.3031 - Identification framework in future networks," available at: <http://www.itu.int/rec/T-REC-Y.3031-201205-P/en>, 2012.

- [Hanka] O. Hanka, C. Spleiss, G. Kunzmann, and J. Eberspächer, "A novel DHTbased network architecture for the next generation internet," Eighth International Conference on Networks, Cancun, Mexico, March 2009.
- [Luo] H. Luo, Y. Qin, and H. Zhang, "A DHT-Based Identifier-to-Locator Mapping Scheme for a Scalable Internet," IEEE Transactions on Parallel and Distributed Systems, October 2009.
- [Ahlgren] B. Ahlgren, J. Arkko, L. Eggert, and J. Rajahalme, "A node identity internetworking architecture," in INFOCOM 2006. 25th IEEE International Conference on Computer Communications Proceedings. Washington, DC, USA: IEEE Computer Society, April 2006, pp. 1-6.
- [Mathy] L. Mathy and L. Iannone, "LISP-DHT: Towards a DHT to map identifiers onto locators," in ReArch'08. Madrid, Spain: ACM, December 2008.
- [Fab1999] Faber, T., Touch, J. and W. Yue, "The TIME-WAIT state in TCP and Its Effect on Busy Servers", Proc. Infocom 1999 pp. 1573-1583.

A.1. Authors' Addresses

Jungha Hong
ETRI
218 Gajeong-ro, Yuseong-gu, Daejeon, Korea

Email: jhong@etri.re.kr

Woojik Chun
Hankuk University of Foreign Studies
81, Oedae-ro, Mohyeon-myeon, Cheoin-gu, Yongin-si, Gyeonggi-do, Korea

Email: woojikchun@gmail.com

Heeyoung Jung
ETRI
218 Gajeong-ro, Yuseong-gu, Daejeon, Korea

Email: hyjung@etri.re.kr

ICNRG
Internet Draft
Intended status: Informational
Expires: Jan 2015

S. Lederer
D. Posch
C. Timmerer
Alpen-Adria University Klagenfurt
C. Westphal, Ed.
Aytac Azgin
Huawei
C. Mueller
Bitmovin
A.Detti
University of Rome Tor Vergata
D. Corujo
University of Aveiro

July 22, 2014

Adaptive Video Streaming over ICN
draft-irtf-icnrg-videostreaming-01.txt

Status of this Memo

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

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79. This document may not be modified, and derivative works of it may not be created, and it may not be published except as an Internet-Draft.

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79. This document may not be modified, and derivative works of it may not be created, except to publish it as an RFC and to translate it into languages other than English.

This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

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/lid-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 January 22, 2014.

Copyright Notice

Copyright (c) 2014 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.

Abstract

This document presents the usage of Information Centric Networks (ICN) for adaptive multimedia streaming and identifies problems, which have to be considered for such applications. Several important topics related to video distribution over ICN are presented, covering a range of scenarios: DASH over ICN, which leverages the recent ISO/IEC MPEG Dynamic Adaptive Streaming over HTTP (DASH) standard, layered encoding over ICN, PPSP over ICN and IPTV over ICN. DASH over ICN offers the possibility to transfer data from multiple sources as well as over multiple links in parallel, which is definitely an important feature, e.g., for mobile devices offering multiple network links. In addition to this, the named multimedia content is routed and cached efficiently by the underlying network. PPSP extends the P2P semantics to video streaming in ICNs. The real time constraints of IPTV and video-conferencing need to be addressed in ICN.

Table of Contents

1. Introduction.....	4
2. Conventions used in this document.....	4
3. Use case scenarios for ICN and Video Streaming.....	5
4. Video streaming and ICN.....	6
4.1. Introduction to client-driven streaming and DASH	6
4.2. Layered Encoding	7
4.3. Interactions of Video Streaming with ICN	7
4.3.1. Interaction of DASH and ICN	7
4.3.2. Interaction of ICN with Layered Encoding	9
4.4. Possible Integration of Video streaming and ICN architecture ..	10
4.4.1. DASH over CCN	10
4.4.2. Testbed, Open Source Tools, and Dataset	12
5. P2P video distribution and ICN.....	13
5.1. Introduction to PPSP	13
5.2. <PPSP over ICN: deployment concepts>	15
5.2.1. PPSP short background	15
5.2.2. From PPSP messages to ICN named-data	15
5.2.3. Support of PPSP interaction through a pull-based ICN API ..	16
5.2.4. Abstract layering for PPSP over ICN	17
5.2.5. PPSP interaction with the ICN routing plane	18
5.2.6. ICN deployment for PPSP	18
5.3. <Impact of MPEG DASH coding schemes>	19
6. IPTV and ICN.....	20
6.1. IPTV challenges	20
6.2. ICN benefits for IPTV delivery	21
7. Future Steps for Video in ICN.....	23
7.1. Heterogeneous Wireless Environment Dynamics	23
7.2. Digital Rights Management of Multimedia Content in ICN	25
8. Security Considerations.....	28
9. IANA Considerations.....	28
10. Conclusions.....	28
11. References.....	29
11.1. Normative References	29
11.2. Informative References	29
12. Authors' Addresses.....	31
13. Acknowledgements.....	32

1. Introduction

The unprecedented growth of video traffic has triggered a rethinking of how content is distributed, both in terms of the underlying Internet architecture and in terms of the streaming mechanisms to deliver video objects.

In particular, the IRTF ICN working group has been chartered to study new architectures centered upon information; the main contributor to Internet traffic (and information dissemination) is video, and this is expected to stay the same in the short- to mid-term future. If ICN is expected to become prominent, it will have to support video streaming efficiently.

As such, it is necessary to discuss along two directions:

- . Can the current video streaming mechanisms be leveraged and adapted to an ICN architecture?
- . Can (and should) new, ICN-specific video streaming mechanisms be designed to fully take advantage of the new abstractions exposed by the ICN architecture?

This document intends to focus on the first question, in an attempt to define the use cases for video streaming and some requirements.

This document focuses on a few scenarios, namely Netflix-like video streaming, peer-to-peer video sharing and IPTV, and identifies how the existing protocols can be adapted to an ICN architecture. In doing so, it also identifies the main issues with these protocols in this ICN context.

Some documents have started to consider the ICN-specific requirements of dynamic adaptive streaming [2][3][4][6].

In this document, we give a brief overview of the existing solutions for the selected scenarios. We then consider the interactions of such existing mechanisms with the ICN architecture and list some of the interactions any video streaming mechanism will have to consider. We then identify some areas for future research.

2. Conventions used in this document

In examples, "C:" and "S:" indicate lines sent by the client and server respectively.

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].

In this document, these words will appear with that interpretation only when in ALL CAPS. Lower case uses of these words are not to be interpreted as carrying RFC-2119 significance.

In this document, the characters ">>" preceding an indented line(s) indicates a compliance requirement statement using the key words listed above. This convention aids reviewers in quickly identifying or finding the explicit compliance requirements of this RFC.

3. Use case scenarios for ICN and Video Streaming

For ICN specific descriptions, we refer to the other working group documents. For our purpose, we assume here that ICN means an architecture where content is retrieved by name and with no binding of content to a specific network location.

The consumption of multimedia content comes along with timing requirements for the delivery of the content, for both, live and on-demand consumption. Additionally, real-time use cases such as audio-/video conferencing [7], game streaming, etc., come along with more strict timing requirements. Long startup delays, buffering periods or poor quality, etc., should be avoided to achieve a good Quality of Experience (QoE) to the consumer of the content. Of course, these requirements are heavily influenced by routing decisions and caching, which are central parts of ICN and which have to be considered when streaming video in such infrastructures.

Due to this range of requirements, we find it useful to narrow the focus on three scenarios (more can be included later):

- a video streaming architecture for playing back movies; this is relevant for the naming and caching aspects of ICN, as well as the interaction with the rate adaptation mechanism necessary to deliver the best QoE to the end-user;
- a peer-to-peer architecture for sharing videos; this introduces more stringent routing requirements in terms of locating copies of the content, as the location of the peers evolves and peers join and leave the swarm they use to exchange video chunks;
- IPTV; this introduces requirements for multicasting and adds stronger delay constraints.

We discuss how the current state-of-the-art protocols in an IP context can be modified for the ICN architecture. For video

streaming, we briefly describe DASH [1], and Layered Encoding (MDC, SVC) and IPTV. For P2P, we describe PPSP. Videoconference and real-time video communications are also part of the scope of this document and will be detailed more in future versions of this document.

4. Video streaming and ICN

4.1. Introduction to client-driven streaming and DASH

Media streaming over the hypertext transfer protocol (HTTP) and in a further consequence streaming over the transmission control protocol (TCP) has become omnipresent in today's Internet. Content providers such as Netflix, Hulu, and Vudu do not deploy their own streaming equipment but use the existing Internet infrastructure as it is and they simply utilize their own services over the top (OTT). This streaming approach works surprisingly well without any particular support from the underlying network due to the use of efficient video compression, content delivery networks (CDNs), and adaptive video players. The assumption of earlier video streaming research, which mostly recommended the user datagram protocol (UDP) and the real time transport protocol (RTP), that it would not be possible to transfer multimedia data smoothly with TCP, because of its throughput variations and large retransmission delays, could be seen as a delusion from today's point of view. HTTP streaming, and especially its most simple form which is known as progressive download, has become very popular over the past few years because it has some major benefits compared to RTP streaming. As a consequence of the consistent use of HTTP for this streaming method, the existing Internet infrastructure, consisting of proxies, caches and CDNs, could be used. Originally, this architecture was designed to support best effort delivery of files and not real time transport of multimedia data. Nevertheless, also real time streaming based on HTTP could take advantage from this architecture, in comparison to RTP, which could not leverage any of the aforementioned components. Another benefit that results from the use of HTTP is that the media stream could easily pass firewalls or network address translation (NAT) gateways, which was definitely a key for the success of HTTP streaming. However, HTTP streaming is not the holy grail of streaming as it also introduces some drawbacks compared to RTP. Nevertheless, in an ICN-based video streaming architecture these aspects also have to be considered.

The basic concept of DASH [1] is to use segments of media content, which can be encoded at different resolutions, bitrates, etc., as so-called representations. These segments are served by conventional HTTP Web servers and can be addressed via HTTP GET requests from the

client. As a consequence, the streaming system is pull-based and the entire streaming logic is located on the client, which makes it scalable, and possible to adapt the media stream to the client's capabilities.

In addition to this, the content can be distributed using conventional CDNs and their HTTP infrastructure, which also scales very well. In order to specify the relationship between the contents' media segments and the associated bitrate, resolution, and timeline, the Media Presentation Description (MPD) is used, which is a XML document. The MPD refers the available media segments using HTTP URLs, which can be used by the client for retrieving them.

4.2. Layered Encoding

Scalable video coding formats the video stream into different layers: a base layer which can be decoded to provide the lowest bit rate for the specific stream, and enhancement layers which can be transmitted separately if network conditions allow. This is used in MPEG-4 scalable profile or H.263+. H264SVC is available, but not much deployed. JPEG2000 has a wavelet transform approach for layered encoding, but has not been deployed much either.

It is not clear if the layered approach is fine-grained enough for rate control.

4.3. Interactions of Video Streaming with ICN

4.3.1. Interaction of DASH and ICN

Video streaming, and DASH in particular, have been designed with goals that are aligned with that of most ICN proposals. Namely, it is a client-based mechanism, which requests items (in this case, chunks of a video stream) by name.

ICN and MPEG-DASH [1] have several elements in common:

- the client-initiated pull approach;
- the content being dealt with in pieces (or chunks);
- the support of efficient replication and distribution of content pieces within the network;
- the session-free nature of the exchange between the client and the server at the streaming layer: the client is free to request any chunk from any location;
- the support for potentially multiple sources.

As ICN is a promising candidate for the Future Internet (FI) architecture, it is useful to investigate its suitability in combination with multimedia streaming standards like MPEG-DASH. In this context, the purpose of this draft is to present the usage of ICN instead of HTTP in MPEG-DASH

However, there are some issues that arise from using a dynamic rate adaptation mechanism in an ICN architecture:

- o Naming of the data in DASH does not necessarily follow the ICN convention of any of the ICN proposals. Several chunks of the same video stream might currently go by different names that for instance do not share a common prefix. There is a need to harmonize the naming of the chunks in DASH with the naming conventions of the ICN. The naming convention of using a filename/time/encoding format could for instance be made compatible with the convention of CCN.
- o While chunks can be retrieved from any server, the rate adaptation mechanism attempts to estimate the available network bandwidth so as to select the proper playback rate and keep its playback buffer at the proper level. Therefore, there is a need to either include some location semantics in the data chunks so as to properly assess the throughput to a specific location; or to design a different mechanism to evaluate the available network bandwidth.
- o The typical issue of access control and accounting happens in this context, where chunks can be cached in the network outside of the administrative control of the content publisher. It might be a requirement from the owner of the video stream that access to these data chunks needs to be accounted/billed/monitored.
- o Dynamic streaming multiplies the representations of a given video stream, therefore diminishing the effectiveness of caching: namely, to get a hit for a chunk in the cache, it has to be for the same format and encoding values. Alternatively, to get the same hit rate as for a stream using a single encoding, the cache size must be scaled up to include all the possible representations.

- o Caching introduces oscillatory dynamics as it may modify the estimation of the available bandwidth between the end user and the repository where it is getting the chunks from. For instance, if an edge cache holds a low resolution representation near the user, the user getting this low resolution chunks will observe a good performance, and will then request higher resolution chunks. If those are hosted on a server with poor performance, then the client would have to switch back to the low representation. This oscillation may be detrimental to the perceived QoE of the user.
- o The ICN transport mechanism needs to be compatible to some extent with DASH. To take a CCN example, the rate at which interests are issued should be such that the chunks received in return arrive fast enough and with the proper encoding to keep the playback buffer above some threshold.
- o The usage of multiple network interfaces is possible in ICN, enabling a seamless handover between them. For the combination with DASH, an intelligent strategy which should focus on traffic load balancing between the available links may be necessary. This would increase the effective media throughput of DASH by leveraging the combined available bandwidth of all links, however, it could potentially lead to high variations of the media throughput.
- o DASH does not define how the MPD is retrieved; hence, this is compatible with CCN. However, the current profiles defined within MPEG-DASH require the MPD to contain HTTP-URLs (incl. http and https URI schemes) to identify segments. To enable a more integrated approach as described in this document, an additional profile for DASH over CCN has to be defined, enabling ICN/CCN-based URIs to identify and request the media segments.

We describe in Section 5 a potential implementation of a dynamic adaptive video stream over ICN, based upon DASH and CCN [5].

4.3.2. Interaction of ICN with Layered Encoding

Issues of interest to an Information-Centric network architecture in the context of layered video streaming include:

- . Caching of the multiple layers. The caching priority should go to the base layer, and defining caching policy to decide when to cache enhancement layers
- . Synchronization of multiple content streams, as the multiple layers may come from different sources in the network (for

instance, the base layer might be cached locally while the enhancement layers may be stored in the origin server)

- . Naming of the different layers: when the client requests an object, the request can be satisfied with the base layer alone, aggregated with enhancement layers. Should one request be sufficient to provide different streams? In a CCN architecture for instance, this would violate a one interest-one data packet principle and the client would need to specify each layer it would like to receive. In a Pub/Sub architecture, the rendezvous point would have to make a decision as to which layers (or which pointer to which layer's location) to return.

4.4. Possible Integration of Video streaming and ICN architecture

4.4.1. DASH over CCN

DASH is intended to enable adaptive streaming, i.e., each content piece can be provided in different qualities, formats, languages, etc., to cope with the diversity of today's networks and devices. As this is an important requirement for Future Internet proposals like CCN, the combination of those two technologies seems to be obvious. Since those two proposals are located at different protocol layers - DASH at the application and CCN at the network layer - they can be combined very efficiently to leverage the advantages of both and potentially eliminate existing disadvantages. As CCN is not based on classical host-to-host connections, it is possible to consume content from different origin nodes as well as over different network links in parallel, which can be seen as an intrinsic error resilience feature w.r.t. the network. This is a useful feature of CCN for adaptive multimedia streaming within mobile environments since most mobile devices are equipped with multiple network links like 3G and WiFi. CCN offers this functionality out of the box which is beneficial when used for DASH-based services. In particular, it is possible to enable adaptive video streaming handling both bandwidth and network link changes. That is, CCN handles the network link decision and DASH is implemented on top of CCN to adapt the video stream to the available bandwidth.

In principle, there are two options to integrate DASH and CCN: a proxy service acting as a broker between HTTP and CCN as proposed in [6], and the DASH client implementing a native CCN interface. The former transforms an HTTP request to a corresponding interest packet as well as a data packet to an HTTP response, including reliable transport as offered by TCP. This may be a good compromise to implement CCN in a managed network and to support legacy devices. As such a proxy is already described in [6] this draft focuses on a more integrated approach, aiming at fully exploiting the potential

CCN DASH Client. That is, a native CCN interface within the DASH client, which adopts a CCN naming scheme (CCN URIs) to denote segments in the Media Presentation Description (MPD). In this architecture, only the network access component on the client has to be modified and the segment URIs within MPD have to be updated according to the CCN naming scheme.

Initially, the DASH client retrieves the MPD containing the CCN URIs of the content representations including the media segments. The naming scheme of the segments may reflect intrinsic features of CCN like versioning and segmentation support. Such segmentation support is already compulsory for multimedia streaming in CCN and, thus, can also be leveraged for DASH-based streaming over CCN. The CCN versioning can be adopted in a further step to signal different representations of the DASH-based content, which enables an implicit adaptation of the requested content to the clients' bandwidth conditions. That is, the interest packet already provides the desired characteristics of a segment (such as bit rate, resolution, etc.) within the content name. Additionally, if bandwidth conditions of the corresponding interfaces or routing paths allow so, DASH media segments could be aggregated automatically by the CCN nodes, which reduces the amount of interest packets needed to request the content. However, such approaches need further research, specifically in terms of additional intelligence and processing power needed at the CCN nodes.

After requesting the MPD, the DASH client will start to request particular segments. Therefore, CCN interest packets are generated by the CCN access component and forwarded to the available interfaces. Within the CCN, these interest packets leverage the efficient interest aggregation for, e.g., popular content, as well as the implicit multicast support. Finally, the interest packets are satisfied by the corresponding data packets containing the video segment data, which are stored on the origin server or any CCN node, respectively. With an increasing popularity of the content, it will be distributed across the network resulting in lower transmission delays and reduced bandwidth requirements for origin servers and content providers respectively.

With the extensive usage of in-network caching, new drawbacks are introduced as a consequence that the streaming logic is located at the client, i.e., clients are not aware of each other and the network infrastructure and cache states. Furthermore, negative effects are introduced when multiple clients are competing for a bottleneck and when caching is influencing this bandwidth competition. As mentioned above, the clients request individual portions of the content based on available bandwidth which is

calculated using throughput estimations. This uncontrolled distribution of the content influences the adaptation process of adaptive streaming clients. The impact of this falsified throughput estimation could be tremendous and leads to a wrong adaptation decision which may impact the Quality of Experience (QoE) at the client, as shown in [8]. In ICN, the client does not have the knowledge from which source the requested content is actually served or how many origin servers of the content are available, as this is transparent and depends on the name-based routing. This introduces the challenge that the adaptation logic of the adaptive streaming client is not aware of the event when the ICN routing decides to switch to a different origin server or content is coming through a different link/interface. As most algorithms implementing the adaptation logic are using bandwidth measurements and related heuristics, the adaptation decisions are no longer valid when changing origin servers (or links) and potentially cause playback interruptions and, consequently, stalling. Additionally, ICN supports the usage of multiple interfaces and a seamless handover between them, which again comes together with bandwidth changes, e.g., switching between fixed and wireless, 3G/4G and WiFi networks, etc. Considering these characteristics of ICN, adaptation algorithms merely based on bandwidth measurements are not appropriate anymore, as potentially each segment can be transferred from another ICN node or interface, all with different bandwidth condition. Thus, adaptation algorithms taking into account these intrinsic characteristics of ICN are preferred over algorithms based on mere bandwidth measurements.

4.4.2. Testbed, Open Source Tools, and Dataset

For the evaluations of DASH over CCN, a testbed with open source tools and datasets is provided in [9]. In particular, it provides two client player implementations, (i) a libdash extension for DASH over CCN and (ii) a VLC plugin implementing DASH over CCN. For both implementations the CCNx implementation has been used as a basis.

The general architecture of libdash is organized in modules, so that the library implements a MPD parser and an extensible connection manager. The library provides object-oriented interfaces for these modules to access the MPD and the downloadable segments. These components are extended to support DASH over CCN and available in a separate development branch of the github project available at <http://www.github.com/bitmovin/libdash>. libdash comes together with a fully featured DASH player with a QT-based frontend, demonstrating the usage of libdash and providing a scientific evaluation platform. As an alternative, patches for the DASH plugin of the VLC player are

provided. These patches can be applied to the latest source code checkout of VLC resulting in a DASH over CCN-enabled VLC player.

Finally, a DASH over CCN dataset is provided in form of a CCNx repository. It includes 15 different quality representation of the well-known Big Buck Bunny Movie, ranging from 100 kbps up to 4500 kbps. The content is split into segments of two seconds, and described by an associated MPD using the presented naming scheme in Section 4.1. This repository can be downloaded from [9], and is also provided by a public accessible CCNx node. Associated routing commands for the CCNx namespaces of the content are provided via scripts coming together with the dataset and can be used as a public testbed.

5. P2P video distribution and ICN

5.1. Introduction to PPSP

P2P video Streaming (PPS) is a popular approach to redistribute live media over Internet. The proposed P2PVS solutions can be roughly classified in two classes:

- Push/Tree based
- Pull/Mesh based

The Push/Tree based solution creates an overlay network among peers that has a tree shape. Using a progressive encoding (e.g. Multiple Description Coding or H.264 Scalable Video Coding), multiple trees could be set up to support video rate adaptation. On each tree an enhancement stream is sent. The more the number of stream received, the higher the video quality. A peer control video rate by fetching or not the streams delivered on the distribution trees.

The Pull/Mesh based solution is inspired by the BitTorrent file sharing mechanism. A Tracker collects information about the state of the swarm (i.e. set of participating peers). A peer forms a mesh overlay network with a subset of peers, and exchange data with them. A peer announces what data items it disposes and requests missing data items that are announced by connected peers. In case of live streaming, the involved data set regards only a recent window of data items published by the source. Also in this case, the use of a progressive encoding can be exploited for video rate adaptation.

Pull/Mesh based P2PVS solutions are the more promising candidate for the ICN deployment, since most of ICN approach provides a pull-based API [5][10][11][12]. In addition, Pull/Mesh based P2PVS are more

robust than Push/Tree based one [13] and the Peer to Peer Streaming Protocol (PPSP) working group [14] is also proposing a Pull/Mesh based solution.

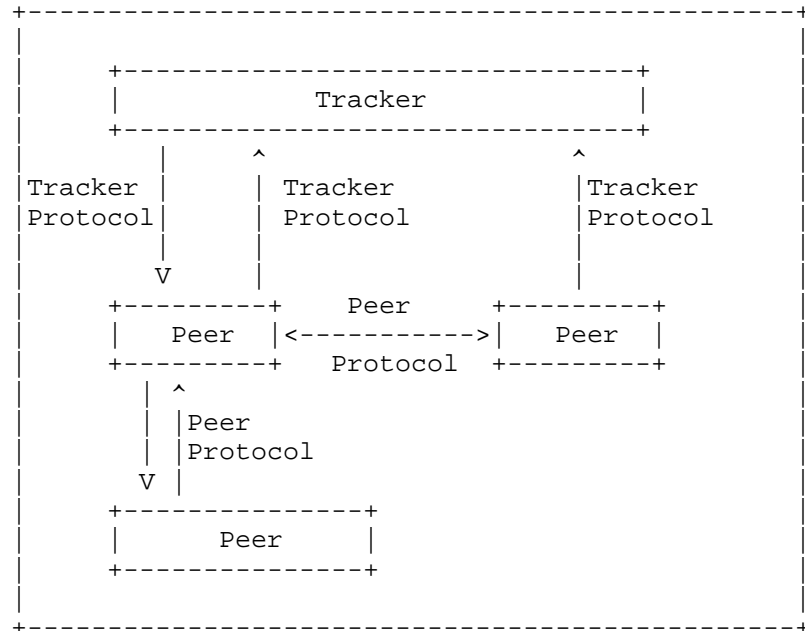


Figure 1: PPSP System Architecture (source [RFC6972])

Figure 1 reports the PPSP architecture presented in [RFC6972]. PEERS announce and share video chunks and a TRACKER maintains a list of PEERS participating in a specific audio/video channel or in the distribution of a streaming file. The tracker functionality may be centralized in a server or distributed over the PEERS. PPSP standardize the Peer and Tracker Protocols, which can run directly over UDP or TCP.

This document discusses some preliminary concepts about the deployment of PPSP on top of an ICN that exposes a pull-based API, meanwhile considering the impact of MPEG DASH streaming format.

5.2. <PPSP over ICN: deployment concepts>

5.2.1. PPSP short background

PPSP specifies peer protocol (PPSPP) [15] and tracker protocol (PPSP-TP)[16].

Some of the operations carried out by the tracker protocol are the followings. When a peer wishes to join the streaming session it contacts the Tracker (CONNECT message), obtains a PEER_ID and a list of PEER_IDs (and IP addresses) of other peers that are participating to the SWARM and that the tracker has singled out for the requesting peer (this may be a subset of the all peers of the SWARM). In addition to this join operation, a peer may contact the tracker to request to renew the list of participating peers (FIND message), to periodically update its status to the tracker (STAT_REPORT message), etc.

Some of the operations carried out by the peer protocol are the following. Using the list of peers delivered by the tracker, a peer establishes a session with them (HANDSHAKE message). A peer periodically announces to neighboring peers which chunks it has available for download (HAVE message). Using these announcements, a peer requests missing chunks from neighboring peers (REQUEST messages), which will send back them (DATA message).

5.2.2. From PPSP messages to ICN named-data

An ICN provides users with data items exposed by names. The bundle name and data item is usually referred as named-data, named-content, etc. To transfer PPSP messages through an ICN the messages should be wrapped as named-data items, and receivers should request them by name.

A PPSP entity receives messages from peers and/or tracker. Some operations require gathering the messages generated by another specific host (peer or tracker). For instance, if a peer A wishes to gain information about video chunks available from peer B, the former shall fetch the PPSP HAVE messages specifically generated by the latter. We refer to these kinds of named-data as "located-named-data", since they should be gathered from a specific location (e.g. peer B).

For other PPSP operations, like to fetch a DATA message (i.e. a video chunk), what it is relevant for a peer is just to receive the requested content, independently from who is the endpoint that

generate the data. We refer this information with the generic term "named-data".

The naming scheme differentiates named-data and located-named-data items. In case of named-data, the naming scheme only includes a content identifier (e.g. the name of the video chunk), without any prefix identifying who provides the content. For instance, a DATA message containing the video chunk n. 1 may be named as "ccnx:/swarmID/chunk/chunkID", where swarmID is a unique identifier of the streaming session, "chunk" is a keyword and chunkID is the chunk identifier (e.g. a integer number).

In case of located-named-data, the naming scheme includes a location-prefix, which uniquely identifies the host generating the data item. This prefix may be the PEER_ID in case the host was a peer or a tracker identifier in case the host was the tracker. For instance, a HAVE message generated by a peer B may be named as "ccnx:/swarmID/peer/PEER_ID/HAVE", where "peer" is a keyword, PEER_ID_B is the identifier of peer B and HAVE is a keyword.

5.2.3. Support of PPSP interaction through a pull-based ICN API

The PPSP procedures are based both on pull and push interactions. For instance, the distribution of chunks availability can be classified as a push-based operation, since a peer sends an "unsolicited" information (HAVE message) to neighboring peers. Conversely the procedure used to receive video chunks can be classified as pull-based, since it is supported by a request/response interaction (i.e. REQUEST, DATA messages).

As we said, we refer to an ICN architecture which provides a pull-based API. Accordingly, the mapping of PPSP pull-based procedure is quite simple. For instance, using the CCN architecture [5] a PPSP DATA message may be carried by a CCN Data message and a REQUEST message can be transferred by a CCN Interest.

Conversely, the support of push-based PPSP operations may be more difficult. We need of an adaptation functionality that carries out a push-based operation using the underlying pull-based service primitives. For instance, a possible approach is to use the request/response (i.e. Interest/Data) four ways handshakes proposed in [7]. Another possibility is that receivers periodically send out request messages of the named-data that neighbors will push and, when available, sender inserts the pushed data within a response message.

5.2.4. Abstract layering for PPSP over ICN

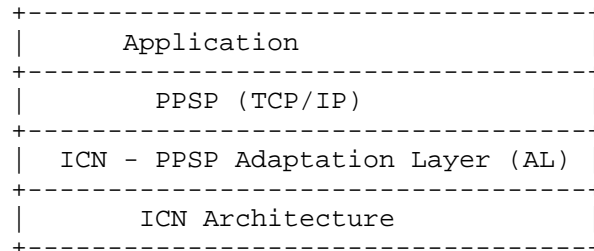


Figure 2: Mediator approach

Figure 2 provides a possible abstract layering for PPSP over ICN. The Adaptation Layer acts as a mediator (proxy) between legacy PPSP entities based on TCP/IP and the ICN architecture. In facts, the role the mediator is to use ICN to transfer PPSP legacy messages.

This approach makes possible to merely reuse TCP/IP P2P applications whose software includes also PPSP functionality. This "all-in-one" development approach may be rather common since the PPSP-Application interface is not going to be specified. Moreover, if the Operating System will provide libraries that expose a PPSP API, these will be initially based on a underlying TCP/IP API. Also in this case, the mediator approach would make possible to easily reuse both the PPSP libraries and the Application on top of an ICN.

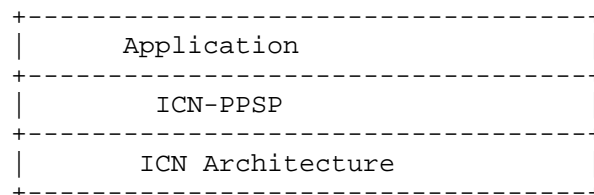


Figure 3: Clean-slate approach

Figure 3 sketches a clean-slate layering approach in which the application directly includes or interacts with a PPSP version based on ICN. Likely such a PPSP_ICN integration could yield a simpler development, also because it does not require implementing a TCP/IP to ICN translation as in the Mediator approach. However, the clean-slate approach requires developing the application (in case of embedded PPSP functionality) or the PPSP library from scratch, without exploiting what might already exist for TCP/IP.

Overall, the Mediator approach may be considered as the first step of a migration path towards ICN native PPSP applications.

5.2.5. PPSP interaction with the ICN routing plane

Upon the ICN API a user (peer) requests a content and the ICN sends it back. The content is gathered by the ICN from any source, which could be the closest peer that disposes of the named-data item, an in-network cache, etc. Actually, "where" to gather the content is controlled by an underlying ICN routing plane, which sets up the ICN forwarding tables (e.g. CCN FIB [5]).

A cross-layer interaction between the ICN routing plane and the PPSP may be required to support a PPSP session. Indeed, ICN shall forward request messages (e.g. CCN Interest) towards the proper peer that can handle them. Depending on the layering approach, this cross-layer interaction is controlled either by the Adaptation Layer or by the ICN-PPSP. For example, if a peer A receives a HAVE message indicating that peer B disposes of the video chunk named "ccnx:/swarmID/chunk/chunkID", then former should insert in its ICN forwarding table an entry for the prefix "ccnx:/swarmID/chunk/chunkID" whose next hop locator (e.g. IP address) is the network address of peer B [17].

5.2.6. ICN deployment for PPSP

The ICN functionality that supports a PPSP session may be "isolated" or "integrated" with the one of a public ICN.

In the isolated case, a PPSP session is supported by an instance of an ICN (e.g. deployed on top of IP), whose functionalities operate only on the limited set of nodes participating to the swarm, i.e. peers and the tracker. This approach resembles the one followed by current P2P application, which usually form an overlay network among peers of a P2P application. And intermediate public IP routers do not carry out P2P functionalities.

In the integrated case, the nodes of a public ICN may be involved in the forwarding and in-network caching procedures. In doing so, the swarm may benefit from the presence of in-network caches so limiting uplink traffic on peers and inter-domain traffic too. These are distinctive advantages of using PPSP over a public ICN, rather than over TCP/IP. In addition, such advantages aren't likely manifested in the case of isolated deployment.

However, the possible interaction between the PPSP and the routing layer of a public ICN may be dramatic, both in terms of explosion of

the forwarding tables and in terms of security. These issues specifically take place for those ICN architectures for which the name resolution (i.e. name to next-hop) occurs en-route, like the CCN architecture.

For instance, using the CCN architecture, to fetch a named-data item offered by a peer A the on-path public ICN entities have to route the request messages towards the peer A. This implies that the ICN forwarding tables of public ICN nodes may contain many entries, e.g. one entry per video chunk, and these entries are difficult to be aggregated since peers avail sparse parts of a big content, whose names have a same prefix (e.g. "ccnx:/swarmID"). Another possibility is to wrap all PPSP messages into a located-named-data. In this case the forwarding tables should contain "only" the PEER_ID prefixes (e.g. "ccnx:/swarmID/peer/PEER_ID"), so scaling down the number of entries from number of chunks to number of peers. However, in this case the ICN mechanisms recognize a same video chunk offered by different peers as different contents, so vanishing caching and multicasting ICN benefits. Moreover, in any case routing entries should be updated either the base of the availability of named-data items on peers or on the presence of peers, and these events in a P2P session is rapidly changing so possibly hampering the convergence of the routing plane. Finally, since peers have an impact on the ICN forwarding table of public nodes, this may open obvious security issues.

5.3. <Impact of MPEG DASH coding schemes>

The introduction of video rate adaptation may valuably decrease the effectiveness of P2P cooperation and of in-network caching, depending of the kind of the video coding used by the MPEG DASH stream.

In case of a MPEG DASH streaming with MPEG AVC encoding, a same video chunk is independently encoded at different rates and the encoding output is a different file for each rate. For instance, in case of a video encoded at three different rates R1,R2,R3, for each segment S we have three distinct files: S.R1, S.R2, S.R3. These files are independent of each other. To fetch a segment coded at R2 kbps, a peer shall request the specific file S.R2. The estimation of the best coding rate is usually handled by receiver-driven algorithms, implemented by the video client.

The independence among files associated to different encoding rates and the heterogeneity of peer bandwidths, may dramatically reduce the interaction among peers, the effectiveness of in-network caching (in case of integrated deployment), and consequently the ability of

PPSP to offload the video server (i.e. a seeder peer). Indeed, a peer A may select a coding rate (e.g. R1) different from the one selected by a peer B (e.g. R2) and this prevents the former to fetch video chunks from the later, since peer B avails of chunks coded at a rate different from the ones needed by A. To overcome this issue, a common distributed rate selection algorithm could force peers to select the same coding rate [17]; nevertheless this approach may be not feasible in the in case of many peers.

The use of SVC encoding (Annex G extension of the H.264/MPEG-4 AVC video compression standard) should make rate adaptation possible, meanwhile neither reducing peer collaborations nor the in-network caching effectiveness. For a single video chunk, a SVC encoder produces different files for the different rates (roughly "layers"), and these files are progressively related each other. Starting from a base-layer which provides the minimum rate encoding, the next rates are encoded as an "enhancement layer" of the previous one. For instance, in case the video is coded with three rates R1 (base-layer), R2 (enhancement-layer n.1), R3 (enhancement-layer n.2), then for each DASH segment we have three files S.R1, S.R2 and S.R3. The file S.R1 is the segment coded at the minimum rate (base-layer). The file S.R2 enhances S.R1, so as S.R1 and S.R2 can be combined to obtain a segment coded at rate R2. To get a segment coded at rate R2, a peer shall fetch both S.R1 and S.R2. This progressive dependence among files that encode a same segment at different rates makes peer cooperation possible, also in case peers player have autonomously selected different coding rates. For instance, if peer A has selected the rate R1, the downloaded files S.R1 are useful also for a peer B that has selected the rate R2, and vice versa.

6. IPTV and ICN

6.1. IPTV challenges

IPTV refers to the delivery of quality content broadcast over the Internet, and is typically associated with strict quality requirements, i.e., with a perceived latency of less than 500 ms and a packet loss rate that is multiple orders lower than the current loss rates experienced in the most commonly used access networks. We can summarize the major challenges for the delivery of IPTV service as follows.

Channel change latency represents a major concern for the IPTV service. Perceived latency during channel change should be less than

500ms. To achieve this objective over the IP infrastructure, we have multiple choices:

- (i) receiving fast unicast streams from a dedicated server (most effective but not resource efficient),
- (ii) connecting to other peers in the network (efficiency depends on peer support, effective and resource efficient, if also supported with a dedicated server),
- (iii) connecting to multiple multicast sessions at once (effective but not resource efficient, and depends on the accuracy of the prediction model used to track user activity).

The second major challenge is the error recovery. Typical IPTV service requirements dictate the mean time between artifacts to be approximately 2 hours. This suggests the perceived loss rate to be around or less than 10^{-7} . Current IP-based solutions rely on the following proactive and reactive recovery techniques: (i) joining the FEC multicast stream corresponding to the perceived packet loss rate (not efficient as the recovery strength is chosen based on worst-case loss scenarios), (ii) making unicast recovery requests to dedicated servers (requires active support from the service provider), (iii) probing peers to acquire repair packets (finding matching peers and enabling their cooperation is another challenge).

6.2. ICN benefits for IPTV delivery

ICN presents significant advantages for the delivery of IPTV traffic. For instance, ICN inherently supports multicast and allows for quick recovery from packet losses (with the help of in-network caching). Similarly, peer support is also provided in the shape of in-network caches that typically act as the middleman between two peers, enabling therefore earlier access to IPTV content.

However, despite these advantages, delivery of IPTV service over Information Centric Networks brings forth new challenges. We can list some of these challenges as follows:

- . Messaging overhead: ICN is a pull-based architecture and relies on a unique balance between requests and responses. A user needs to make a request for each data packet. In the case of IPTV, with rates up to, and likely to be, above 15Mbps, we observe significant traffic upstream to bring those streams.

As the number of streams increase (including the same session at different quality levels), so as the burden on the routers. Even if the majority of requests are aggregated at the core, routers close to the edge (where we observe the biggest divergence in user requests) will experience a significant increase in overhead to process these requests. The same is true at the user side, as the uplink usage multiplies in the number of sessions a user requests (for instance, to minimize the impact of bandwidth fluctuations).

- . Cache control: As the IPTV content expires at a rapid rate (with a likely expiry threshold of 1s), we need solutions to effectively flush out such content to also prevent degradatory impact on other cached content, with the help of intelligently chosen naming conventions. However, to allow for fast recovery and optimize access time to sessions (from current or new users), the timing of such expirations needs to be adaptive to network load and user demand. However, we also need to support quick access to earlier content, whenever needed, for instance, when the user accesses the rewind feature (note that in-network caches will not be of significant help in such scenarios due to overhead required to maintain such content).
- . Access accuracy: To receive the up-to-date session data, users need to be aware of such information at the time of their request. Unlike IP multicast, since the users join a session indirectly, session information is critical to minimize buffering delays and reduce the startup latency. Without such information, and without any active cooperation from the intermediate routers, stale data can seriously undermine the efficiency of content delivery. Furthermore, finding a cache does not necessarily equate to joining a session, as the look-ahead latency for the initial content access point may have a shorter lifetime than originally intended. For instance, if the user that has initiated the indirect multicast leaves the session early, the requests from the remaining users need to experience an additional latency of one RTT as they travel towards the content source. If the startup latency is chosen depending on the closeness to the intermediate router, going to the content source in-session can lead to undesired pauses.

7. Future Steps for Video in ICN

The explosion of online video services, along with their increased consumption by mobile wireless terminals, further exacerbates the challenges of Video Adaptation leveraging ICN mechanisms. The following sections present a series of research items derived from these challenges, further introducing next steps for the subject.

7.1. Heterogeneous Wireless Environment Dynamics

With the ever-growing increase in online services being accessed by mobile devices, operators have been deploying different overlapping wireless access networking technologies. In this way, in the same area, user terminals are within range of different cellular, Wi-Fi or even WiMAX networks. Moreover, with the advent of the Internet of Things (e.g., surveillance cameras feeding video footage), this list can be further complemented with more specific short-range technologies, such as Bluetooth or ZigBee.

In order to leverage from this plethora of connectivity opportunities, user terminals are coming equipped with different wireless access interfaces, providing them with extended connectivity opportunities. In this way, such devices become able to select the type of access which best suits them according to different criteria, such as available bandwidth, battery consumption, access to different link conditions according to the user profile or even access to different content. Ultimately, these aspects contribute to the Quality of Experience perceived by the end-user, which is of utmost importance when it comes to video content.

However, the fact that these users are mobile and using wireless technologies, also provides a very dynamic setting, where the current optimal link conditions at a specific moment might not last or be maintained while the user moves. These aspects have been amply analyzed in recently finished projects such as FP7 MEDIEVAL [18], where link events reporting on wireless conditions and available alternative connection points were combined with video requirements and traffic optimization mechanisms, towards the production of a joint network and mobile terminal mobility management decision. Concretely, in [19] link information about the deterioration of the wireless signal was sent towards a mobility management controller in the network. This input was combined with information about the user profile, as well as of the current video service requirements, and used to trigger the decrease or increase of scalable video layers, adjusting the video to the ongoing link conditions. Incrementally,

the video could also be adjusted when a new better connectivity opportunity presents itself.

In this way, regarding Video Adaptation, ICN mechanisms can leverage from their intrinsic multiple source support capability and go beyond the monitoring of the status of the current link, thus exploiting the availability of different connectivity possibilities (e.g., different "interfaces"). Moreover, information obtained from the mobile terminal's point of view of its network link, as well as information from the network itself (i.e., load, policies, and others), can generate scenarios where such information is combined in a joint optimization procedure allowing the content to be forward to users using the best available connectivity option (e.g., exploiting management capabilities supported by ICN intrinsic mechanisms as in [20]).

In fact, ICN base mechanisms can further be exploited in enabling new deployment scenarios such as preparing the network for mass requests from users attending a large multimedia event (i.e., concert, sports), allowing video to be adapted according to content, user and network requirements and operation capabilities in a dynamic way.

The enablement of such scenarios require further research, with the main points highlighted as follows:

- . Development of a generic video services (and obviously content) interface allowing the definition and mapping of their requirements (and characteristics) into the current capabilities of the network;
- . How to define a scalable mechanism allowing either the video application at the terminal, or some kind of network management entity, to adapt the video content in a dynamic way;
- . How to develop the previous research items using intrinsic ICN mechanisms (i.e., naming and strategy layers);
- . Leverage intelligent pre-caching of content to prevent stalls and poor quality phases, which lead to bad Quality of Experience of the user. This includes in particular the usage in mobile environments, which are characterized by severe bandwidth changes as well as connection outages, as shown in [21].

7.2. Digital Rights Management of Multimedia Content in ICN

This subsection discusses the need for Digital Rights Management (DRM) functionalities for multimedia streaming over ICN. The discussion will show that Broadcast Encryption (BE) is a suitable basis for DRM functionalities in conformance to the ICN communication paradigm. Especially when network inherent caching is considered the advantage of BE will be highlighted.

It is assumed that ICN will be used heavily for digital content dissemination. When digital content is distributed it is vital to consider DRM. In today's Internet there are two predominant classes of business models for on-demand video streaming. The first model is based on advertising revenues. Non copyright protected usually user-generated content (UGC) is offered by large infrastructure providers like Google (YouTube) at no charge. The infrastructure is financed by spliced advertisements into the content. In this context DRM considerations are usually not required, since producers of UGC just strive for the maximum possible dissemination. Producers of UGC are mainly interested to share content with their families, friends, colleges or others and have no intention to make profit. However, the second class of business models requires DRM, because they are primarily profit oriented. For example, large on-demand streaming platforms like Netflix establish business models based on subscriptions. Consumers have to pay a monthly fee in order to get access to copyright protected content like TV series, movies or music. From the perspective of the service providers and the copyright owners only clients that pay the fee should be able to access and consume the content. Anyway, the challenge is to find an efficient and scalable way of access control to digital content, which is distributed in information-centric networks.

In ICN, data packets can be cached inherently in the network and any network participant can request a copy of these packets. This makes it very difficult to implement an access control for content that is distributed via ICN. A naive approach is to encrypt the transmitted data for each consumer with a distinct key. This hinders everyone else than the intended consumers to decrypt and consume the data. However, this approach is not suitable for ICN's communication paradigm since it would destruct any benefits gained from network inherent caching. Even if multiple consumers request the same content the requested data for each consumer would differ using this approach. A better but still insufficient idea is to use a single key for all consumers. This does not destruct the benefits of ICN's caching ability. Though, the drawback is that if one of the consumers illegally distributes the key the system is broken and any entity in the network can access the data. Changing the key after

such an event is useless since the provider has no possibility to identify the illegal distributor. Therefore this person cannot be stopped from distributing the new key again. In addition to this issue other challenges have to be considered. Subscriptions expire after a certain time and then it has to be ensured that these consumers cannot access the content anymore. For a provider that daily serves millions of consumers (e.g. Netflix) there could be a significant number of expiring subscriptions a day. Publishing a new key every time a subscription expires would require an unsuitable amount of computational power just to re-encrypt the collection of audio-visual content.

A possible approach to solve these challenges is Broadcast Encryption (BE) [BE] as proposed in [DAECC]. The ongoing discussion in this subsection will focus only on BE as an enabler for DRM functionality in the use case of ICN video streaming. This subsection continues with the explanation of how BE works and shows how BE can be used to implement an access control scheme in the context of content distribution in ICN.

BE actually carries a misleading name. One might expect a concrete encryption scheme. However, it belongs to the family of key-management schemes (KMS). KMS are responsible for the generation, exchange, storage and replacement of cryptographic keys. The most interesting characteristics of Broadcast Encryption Schemes (BES) are:

- . A BES typically uses a global trusted entity called the licensing agent (LA), which is responsible for spreading a set of pre-generated secrets among all participants. Each participant gets a distinct subset of secrets assigned from the LA.
- . The participants can agree on a common session key, which is chosen by the LA. The LA broadcasts an encrypted message that includes the key. Participants with a valid set of secrets can derive the session-key from this message.
- . The number of participants in the system can change dynamically. Entities may join or leave the communication group at any time. If a new entity joins the LA passes on a valid set of secrets to that entity. If an entity leaves (or is forced to leave) the LA revokes the entity's subset of keys, which means that it cannot derive the correct session key anymore when a new key is distributed by the LA.
- . -Traitors (entities that reveal their secrets) can be traced and excluded from ongoing communication. The algorithms and preconditions to identify a traitor vary between concrete BES.

This listing already illustrates why BE is suitable to control the access to data that is distributed via an information-centric network. BE enables the usage of a single session key for confidential data transmission between a dynamically changing subset or network participants. ICN caches can be utilized since the data is encrypted only with a single key known by all legitimate clients. Furthermore, traitors can be identified and removed from the system. The issue of re-encryption still exists, because the LA will eventually update the session key when a participant should be excluded. However, this disadvantage can be relaxed in some way if the following points are considered:

- . The updates of the session key can be delayed until a set of compromised secrets has been gathered. Note that secrets may become compromised because of two reasons. First, if the secret has been illegally revealed by a traitor. Second, if the subscription of an entity expires. Delayed revocation temporarily enables some non-legitimate entities to consume content. However, this should not be a severe problem in home entertainment scenarios. Updating the session key in regular (not too short) intervals is a good tradeoff. The longer the interval last the less computational resources are required for content re-encryption and the better the cache utilization in the ICN will be. To evict old data from ICN caches that has been encrypted with the prior session key the publisher could indicate a lifetime for transmitted packets.
- . Content should be re-encrypted dynamically at request time. This has the benefit that untapped content is not re-encrypted if the content is not requested during two session key updates and therefore no resources are wasted. Furthermore, if the updates are triggered in non-peak times the maximum amount of resource needed at one point in time can be lowered effectively, since in peak times generally more diverse content is requested.
- . Since the amount of required computational resources may vary strongly from time to time it would be beneficial for any streaming provider to use cloud-based services to be able to dynamically adapt the required resources to the current needs. Regarding to a lack of computation time or bandwidth the cloud service could be used to scale up to overcome shortages.

Figure 4 show the potential usage of BE in a multimedia delivery frameworks that builds upon ICN infrastructure and uses the concept of dynamic adaptive streaming, e.g., DASH. BE would be implemented on the top to have an efficient and scalable way of access control to the multimedia content.

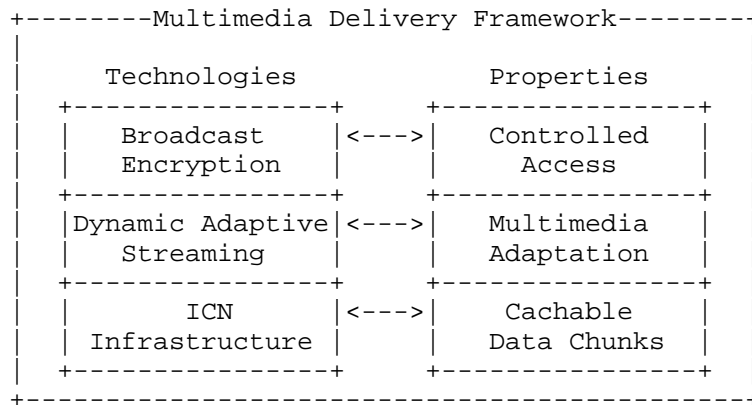


Figure 4: A potential multimedia framework using BE.

8. Security Considerations

This is informational. Security considerations are TBD.

9. IANA Considerations

This is informational. IANA considerations are TBD.

10. Conclusions

This draft proposed adaptive video streaming for ICN, identified potential problems and presented the combination of CCN with DASH as a solution. As both concepts, DASH and CCN, maintain several elements in common, like, e.g., the content in different versions being dealt with in segments, combination of both technologies seems useful. Thus, adaptive streaming over CCN can leverage advantages such as, e.g., efficient caching and intrinsic multicast support of CCN, routing based on named data URIs, intrinsic multi-link and multi-source support, etc.

In this context, the usage of CCN with DASH in mobile environments comes together with advantages compared to today's solutions, especially for devices equipped with multiple network interfaces. The retrieval of data over multiple links in parallel is a useful feature, specifically for adaptive multimedia streaming, since it offers the possibility to dynamically switch between the available links depending on their bandwidth capabilities, transparent to the actual DASH client.

11. References

11.1. Normative References

- [RFC6972] Y. Zhang, N. Zong, "Problem Statement and Requirements of the Peer-to-Peer Streaming Protocol (PPSP)", RFC6972, July 2013

11.2. Informative References

- [1] ISO/IEC DIS 23009-1.2, Information technology - Dynamic adaptive streaming over HTTP (DASH) - Part 1: Media presentation description and segment formats
- [2] Lederer, S., Mueller, C., Rainer, B., Timmerer, C., Hellwagner, H., "An Experimental Analysis of Dynamic Adaptive Streaming over HTTP in Content Centric Networks", in Proceedings of the IEEE International Conference on Multimedia and Expo 2013, San Jose, USA, July, 2013
- [3] Liu, Y., Geurts, J., Point, J., Lederer, S., Rainer, B., Mueller, C., Timmerer, C., Hellwagner, H., "Dynamic Adaptive Streaming over CCN: A Caching and Overhead Analysis", in Proceedings of the IEEE international Conference on Communication (ICC) 2013 - Next-Generation Networking Symposium, Budapest, Hungary, June, 2013
- [4] Grandl, R., Su, K., Westphal, C., "On the Interaction of Adaptive Video Streaming with Content-Centric Networks", eprint arXiv:1307.0794, July 2013.
- [5] V. Jacobson, D. Smetters, J. Thornton, M. Plass, N. Briggs and R. Braynard, "Networking named content", in Proc. of the 5th int. Conf. on Emerging Networking Experiments and Technologies (CoNEXT '09). ACM, New York, NY, USA, 2009, pp. 1-12.
- [6] A. Detti, M. Pomposini, N. Blefari-Melazzi, S. Salsano and A. Bragagnini, "Offloading cellular networks with Information-Centric Networking: The case of video streaming", In Proc. of the Int. Symp. on a World of Wireless, Mobile and Multimedia Networks (WoWMoM '12), IEEE, San Francisco, CA, USA, 1-3, 2012.
- [7] V. Jacobson, D. K. Smetters, N. H. Briggs, M. F. Plass, P. Stewart, J. D. Thornton, and R. L. Braynard, "VoCCN: Voice over content-centric networks," in ACM ReArch Workshop, 2009

- [8] Christopher Mueller, Stefan Lederer and Christian Timmerer, A proxy effect analysis and fair adaptation algorithm for multiple competing dynamic adaptive streaming over HTTP clients, In Proceedings of the Conference on Visual Communications and Image Processing (VCIP) 2012, San Diego, USA, November 27-30, 2012.
- [9] DASH Research at the Institute of Information Technology, Multimedia Communication Group, Alpen-Adria Universitaet Klagenfurt, URL: <http://dash.itec.aau.at>
- [10] A. Detti, N. Blefari-Melazzi, S. Salsano, and M. Pomposini, "CONET: A content centric inter-networking architecture," in ACM Workshop on Information-Centric Networking (ICN), 2011.
- [11] W. K. Chai, N. Wang, I. Psaras, G. Pavlou, C. Wang, G. C. de Blas, F. Ramon-Salguero, L. Liang, S. Spirou, A. Beben, and E. Hadjioannou, "CURLING: Content-ubiquitous resolution and delivery infrastructure for next-generation services," IEEE Communications Magazine, vol. 49, no. 3, pp. 112-120, March 2011
- [12] NetInf project Website <http://www.netinf.org>
- [13] N. Magharei, R. Rejaie, Yang Guo, "Mesh or Multiple-Tree: A Comparative Study of Live P2P Streaming Approaches," INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE , vol., no., pp.1424,1432, 6-12 May 2007
- [14] PPSP WG Website <https://datatracker.ietf.org/wg/ppsp/>
- [15] A. Bakker, R. Petrocco, V. Grishchenko, "Peer-to-Peer Streaming Peer Protocol (PPSPP)", draft-ietf-ppsp-peer-protocol-08
- [16] Rui S. Cruz, Mario S. Nunes, Yingjie Gu, Jinwei Xia, Joao P. Taveira, Deng Lingli, "PPSP Tracker Protocol-Base Protocol (PPSP-TP/1.0)", draft-ietf-ppsp-base-tracker-protocol-02
- [17] A.Detti, B. Ricci, N. Blefari-Melazzi, "Peer-To-Peer Live Adaptive Video Streaming for Information Centric Cellular Networks", IEEE PIMRC 2013, London, UK, 8-11 September 2013
- [18] <http://www.ict-medieval.eu>

- [19] B. Fu, G. Kunzmann, M. Wetterwald, D. Corujo, R. Costa, "QoE-aware Traffic Management for Mobile Video Delivery", Proc. 2013 IEEE ICC, Workshop on Immersive & Interactive Multimedia Communications over the Future Internet (IIMC), Budapest, Hungary, Jun 2013.
- [20] Daniel Corujo, Ivan Vidal, Jaime Garcia-Reinoso, Rui L. Aguiar, "A Named Data Networking Flexible Framework for Management Communications", IEEE Communications Magazine, Vol. 50, no. 12, pp. 36-43, Dec 2012
- [21] Barry Crabtree, Tim Stevens, Brahin Allan, Stefan Lederer, Daniel Posch, Christopher Mueller, Christian Timmerer, Video Adaptation in Limited or Zero Network Coverage, CCNxConn 2013,PARC, Palo Alto, pp. 1-2, 2013
- [22] Fiat, A., Naor, M., "Broadcast Encryption", in Advances in Cryptology (Crypto'93), volume 773 of Lecture Notes in Computer Science, pages 480-491. Springer Berlin / Heidelberg, 1994.
- [23] Posch, D., Hellwagner, H., Schartner, P., "On-Demand Video Streaming based on Dynamic Adaptive Encrypted Content Chunks", in Proceedings of the 8th International Workshop on Secure Network Protocols (NPSec' 13), Los Alamitos, IEEE Computer Society Press, October, 2013.

12. Authors' Addresses

Stefan Lederer, Christian Timmerer, Daniel Posch
Alpen-Adria University Klagenfurt
Universitaetsstrasse 65-67, 9020 Klagenfurt, Austria

Email: {firstname.lastname}@itec.aau.at

Cedric Westphal, Aytac Azgin
Huawei
2330 Central Expressway, Santa Clara, CA95050, USA

Email: {first.last}@huawei.com

Christopher Mueller
bitmovin GmbH
Lakeside B01, 9020 Klagenfurt, Austria

Email: christopher.mueller@bitmovin.net

Andrea Detti
Electronic Engineering Dept.
University of Rome Tor Vergata
Via del Politecnico 1, Rome, Italy

Email: andrea.detti@uniroma2.it

Daniel Corujo,
Advanced Telecommunications and Networks Group
Instituto de Telecomunicacoes
Campus Universitario de Santiago
P-3810-193 Aveiro, Portugal

Email: dcorujo@av.it.pt

13. Acknowledgements

This work was supported in part by the EC in the context of the SocialSensor (FP7-ICT-287975) project and partly performed in the Lakeside Labs research cluster at AAU. SocialSensor receives research funding from the European Community's Seventh Framework Programme. The work for this document was also partially performed in the context of the FP7/NICT EU-JAPAN GreenICN project, <http://www.greenicn.org>. Apart from this, the European Commission has no responsibility for the content of this draft. The information in this document is provided as is and no guarantee or warranty is given that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability.

ICNRG
Internet-Draft
Intended status: Informational
Expires: January 4, 2015

M. Arumaithurai
J. Chen
X. Fu
University of Goettingen
K. Ramakrishnan
University of California, Riverside
J. Seedorf
NEC
July 3, 2014

Enabling Publish/Subscribe in ICN
draft-jiachen-icn-pubsub-00

Abstract

Information-Centric Networks (ICN) provide substantial flexibility for users to obtain information without regard to the source of the information or its current location. Publish/subscribe (pub/sub) systems have gained popularity in society to provide the convenience of removing the temporal dependency of the user having to indicate an interest each time he or she wants to receive a particular piece of related information. Such an "information-centric" communication model should be supported in the new ICN network paradigm. This document outlines some research directions for ICN with respect to enhancing the inherently pull-based ICN approaches for achieving efficient pub/sub capability.

Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 4, 2015.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. Pub/Sub Communication	3
3. Scenarios of Pub/Sub Architecture	4
3.1. Online Social Networks and RSS Feeds	4
3.2. Online Gaming and Audio/Video Conferencing	4
3.3. Notification Systems in Disaster	5
4. Requirements of an Efficient Pub/Sub Architecture	5
5. Related Work	7
5.1. IP/Overlay Multicast	7
5.2. Named-Data Networking (NDN)	7
5.3. Content-Oriented Publish/Subscribe(COPSS)	8
5.4. Other Related Works	8
6. Standardisation Considerations	9
7. References	10
7.1. Normative References	10
7.2. Informative References	10
Appendix A. Acknowledgment	11
Authors' Addresses	11

1. Introduction

This document points out the need to support publish/subscribe (pub/sub) capabilities in ICN and the problems with the existing solutions. Further, the document discusses potential directions for enhancing Information Centric Networking (ICN) to achieve efficient pub/sub.

Section 2 describes the pub/sub systems and the challenges of such systems to the current Internet. Section 3 demonstrates the use of pub/sub systems in different scenarios. Section 4 outlines the requirements of an efficient pub/sub architecture and Section 5

discusses the related works and some possible shortcomings. In Section 6 we brief our standardisation considerations.

2. Pub/Sub Communication

Users increasingly desire access to information, ranging from news, financial markets, healthcare, to disaster relief and beyond, independent of who published it, where it is located, and often, when it was published. Typical representation of these usages are microblogs, RSS feed, social network, search engines, etc. A consumer may not wish (or it may even be infeasible) to receive all of the "channels" belonging to a myriad of information providers that disseminate items of interest, either on demand (such as web, twitter, blogs and social networks), or tune to a broadcast channel (e.g., television, radio, newspaper). In these cases, the consumer would rather prefer obtaining the data based on Content Descriptors (CD) such as a keyword, a tag, or a property of the content (publisher identity, published date etc.).

Publish/subscribe (pub/sub) systems are particularly suited for such kind of large scale content-oriented information dissemination, and provide the flexibility for users to subscribe to information of interest, without being intimately tied to when that information is made available by publishers. With the use of an appropriate interface, users can select and filter the information desired so that they receive only what they are interested in, often irrespective of the publisher.

Intelligent end-systems and information aggregators (e.g., Google News and Yahoo! News, cable and satellite providers) have increasingly adapted their interfaces to provide a content-oriented pub/sub-based delivery method. However, these mechanisms are built on top of a centralized server based framework and can also result in a waste of network resources as shown in [Ramasubramanian2006][Katsaros2011], since the Internet protocol suite is focused on end-to-end delivery of data. Furthermore, issues of "coverage" and "timeliness" still exist in such forms of dissemination, where the aggregator may be selective in what information is made available.

Information-Centric Networks (ICN) is a new network paradigm that intends to achieve large scale data delivery with greater ease for users, greater scalability in terms of the amount of information disseminated as well as number of producers and consumers of information, and greater efficiency in terms of network and server resource utilization.

It is also desirable for such a network to assist the pub/sub communication model that delivers the information from any of the producers to all subscribers. Moreover, it is desirable for the network to assist in delivering fine-grained information to the subscriber.

Recently, works such as [Schmidt2012],[Carzaniga2011],[Chen2011],[Chen2012] have also highlighted the need for ICN to support a pub/sub like communication model.

3. Scenarios of Pub/Sub Architecture

In this section, we list several use cases of pub/sub architectures in ICN. They help us to understand the requirements of an efficient pub/sub architecture and why the existing solutions fall short.

3.1. Online Social Networks and RSS Feeds

Online social networks (e.g., Twitter, Facebook, etc.) and Rich Site Summary (RSS) feeds are typical use cases for a content-centric pub/sub system. In such systems, the receivers receive messages either from friends, followees, or from some information aggregators. They do not care which exact machine is sending the message (content-centric), nor do they know when and what is the name of the next message they are going to receive (temporal separation).

To prevent the receivers from polling all the possible providers, existing systems use web servers as rendezvous points: the publishers send new messages to the servers and the receivers/subscribers poll the server periodically. This still causes great wastage for the (HTTP) servers answering "304 - Not Modified" repeatedly since the message update frequency is usually lower than the polling frequency.

3.2. Online Gaming and Audio/Video Conferencing

Massively multiplayer online role-playing games (MMORPGs, e.g., Counter-Strike, Quake, World of Warcraft, etc.) and audio/video conferencing (e.g., Skype meeting, Web Whiteboard, Etherpad, etc.) is another kind of content-centric pub/sub systems. Similar to the social network scenario, users in such systems only care about the content, either the area of interest (AoI) or the conference partners, and they do not know when and from where the next message will come. But different from the previous scenario, such systems require real-time update (message) delivery and these messages are usually smaller in size compared to the online social networks.

Many of these systems choose to use HTTPS or direct TCP connection between the server and the users to enable the capability of server "pushing" the updates to the user. But maintaining such links are costly. MMORPGs usually limit the number of players in a same game which greatly reduces the interesting of these games.

3.3. Notification Systems in Disaster

Disasters have often disrupted communications because of damages to critical infrastructure. For instance in the aftermath of the Japanese Earthquake in 2011, approximately 1,200,000 fixed telephone lines and 15,000 base-stations were not functioning. On average, 22% (with peaks up to 65% in some areas) of the base-stations had to shut down due to the lack of power or damages to the infrastructure.

Contradictory to the loss of available hardware capacity, during and in the aftermath of a disaster, there is a substantial increase in the amount of traffic generated because of the natural anxiety and panic among people and the need to organize rescue and emergency services. Many of these traffic are in the form of a pub/sub communication model, e.g., the government needs to publish some notifications (recovery status, new shelter locations, etc.), the refugees need to notify their friends about their safety, or people needs to ask for help from ambulances or fire brigade. In the Japanese case, the congestion caused by such traffic resulted in restrictions in voice traffic up to 95%, including emergency priority calls.

4. Requirements of an Efficient Pub/Sub Architecture

Given a pub/sub communication model as described in Section 2, on a high-level one can derive the following (incomplete) list of basic requirements:

- o Push enabled dissemination: To ensure that subscribers receive information in a timely manner, the target system must provide the ability for publishers to push information to online subscribers interested in it. Such timely dissemination is useful in many scenarios such as disaster (e.g., Tsunami) warnings, stock market information, news and gaming.
- o Decouple publishers and subscribers: As the number of publishers and subscribers increases, it is important for the network to be content-centric (using content names rather than addresses for routing), while still providing the appropriate association between them (publishers need not know who the subscribers are, and vice versa). Furthermore, each subscriber may be a publisher

as well (e.g., Twitter allows users to be both subscribers and publishers of data).

- o Scalability: The target system must handle a large number of publishers and subscribers. Minimizing the amount of state maintained in the network, ensuring the load on the publisher grows slowly (sub-linearly) with the number subscribers, the load on subscribers also grows slowly with the number of publishers (e.g., dealing with the burden of duplicate elimination). Importantly, the load on the network should not grow significantly with the growth in the number of publishers and subscribers. We also recognize the need to accommodate a very large range in the amount of information that may be disseminated, and the need for all elements of the target system in a content centric environment to scale in a manageable way.
- o Efficiency: The system must utilize network and server resources efficiently. It is desirable that content is not transmitted multiple times by a server or on a link. Furthermore, the overhead on publisher and subscriber end-points to query unnecessarily for information must be minimized.

Additionally, to support a full-fledge pub-sub environment, it is desirable that the target system support the following additional features:

- o Support hierarchies and context in naming content: We believe it is desirable to be able to exploit both context and hierarchies in identifying content. Hierarchical naming has been recognized by NDN as well. Exploiting context enables a richer identification of content (in both subscriptions and published information), as noted in the database community (and adopted in [Fenner2005]).
- o Supporting two-step dissemination for policy control and efficiency: We recognize the need for pub/sub environments to support a two-step dissemination process both for reasons of policy and access control at the publisher as well as managing delivery of large volume content. In such a scenario, the target system would be designed to publish only a snippet of the data (containing a description of the content and the method how to obtain it) to subscribers.
- o Subscriber offline support: Another typical characteristic of pub-sub environments is that subscribers could be offline at the time the data is published. There is clearly a need for asynchronous delivery of information in a pub/sub environment in an efficient, seamless and scalable manner. The system needs to allow users who were online to retrieve the data that they have missed. It should

also allow new subscribers to retrieve previously published content that they are interested in. We envisage a server that stores all the content published.

5. Related Work

5.1. IP/Overlay Multicast

IP multicast [RFC1112] is a candidate solution for efficiently delivering content to multiple receivers. A sender sends data to a multicast group address that subscribers could join. Multicast routing protocols such as PIM-SM [RFC4601] construct and maintain a tree from each sender to all receivers of a multicast group. However, IP multicast isn't an efficient pub/sub delivery mechanism for several reasons: 1) IP multicast is designed for delivery of packets to connected end-points. Dealing with disconnected operation (when subscribers are online) would have to be an application layer issue. Overlay multicast solutions such as [Jannotti2000][Chu2002][Banerjee2002] are agnostic of the underlying network topology, usually relying on multiple unicasts in the underlay path and are therefore also inefficient as a pub/sub delivery mechanism. 2) The somewhat limited multicast group address space makes it difficult to support a direct mapping of CDs to IP multicast addresses. 3) Current IP multicast is not able to exploit relationships between information elements, such as CDs. CDs may be hierarchical or may have a contextual relationship, which enables multiple CDs to be mapped to a group. For example, consider a publisher that sends a message to all the subscribers interested in football, and subscribers who are interested in receiving messages about all sports. The message from the publisher will have to be sent to two distinct IP multicast groups. If there happens to be a subscriber of messages on sports and football, (s)he will receive the same message twice and will have to perform redundancy elimination in the application layer. The result is a waste in network traffic and processing at both ends.

5.2. Named-Data Networking (NDN)

CCN/NDN has limited intrinsic support for pub/sub systems, a critical need in a content centric environment. The aggregation of pending Interests at routers achieves efficient dissemination of information from NDN nodes. But this aggregation is similar to a cache hit in a content distribution network (CDN) cache, which occurs only if subscribers send their Interests with some temporal locality. Thus it avoids multiple Interest queries having to be processed directly by the content provider. Note however that this is still a pull-based information delivery method and depends both on temporal locality of interests and a large enough cache to achieve effective

caching in the (content centric) network. On the other hand, native multicast support allows for a much more scalable push-based pub/sub environment, since it is not sensitive to issues such as the cycling of the cache when a large amount of information is disseminated.

5.3. Content-Oriented Publish/Subscribe(COPSS)

COPSS enhances CCN/NDN with a push-based delivery mechanism using multicast in a content-centric framework. It is designed to satisfy the requirements mentioned above, especially to provide temporal separation between subscription (or expression of Interest) and publication. At the content-centric network layer, COPSS uses a multiple-sender, multiple-receiver multicast capability, in much the same manner as PIM-SM.

5.4. Other Related Works

Here we list the other related works we are considering. The list might not be complete and we intend to add to it based on feedback received in further revisions.

- o A. Carzaniga, M. Rutherford, A. Wolf, A routing scheme for content-based networking, in: INFOCOM, 2004.
- o B. Segall, D. Arnold, J. Boot, M. Henderson, T. Phelps, Content Based Routing with Elvin, in: AUUG2K, 2000.
- o C. Esteve, F. Verdi, M. Magalhaes, Towards a new generation of information-oriented Internetworking architectures, in: ReArch, 2008.
- o G. Chockler, R. Melamed, Y. Tock, R. Vitenberg, SpiderCast: a scalable interest-aware overlay for topic-based pub/sub communication, in: DEBS, 2007.
- o H. Eriksson, Mbone: the multicast backbone, Commun. ACM 37 (8) (1994) 54-60.
- o M. Ott, L. French, R. Mago, D. Makwana, Xml-based semantic multicast routing: an overlay network architecture for future information services, in: GLOBECOM, 2004.
- o P. T. Eugster, P. A. Felber, R. Guerraoui, A.-M. Kermarrec, The many faces of publish/subscribe, ACM Comput. Surv. 35 (2) (2003) 114-131.

- o R. Baldoni, R. Beraldi, V. Quema, L. Querzoni, S. Tucci-Piergiovanni, TERA: topic-based event routing for peer-to-peer architectures, in: DEBS, 2007.
- o R. V. Renesse, K. P. Birman, W. Vogels, Astrolabe: A Robust and Scalable Technology for Distributed System Monitoring, Management, and Data Mining, ACM TOCS 21 (2001) 66-85.
- o S. Voulgaris, E. Riviere, A.-M. Kermarrec, M. Van Steen, Sub-2-Sub: Self-Organizing Content-Based Publish and Subscribe for Dynamic and Large Scale Collaborative Networks, Research report, INRIA (December 2005).
- o T. Koponen, M. Chawla, B.-G. Chun, A. Ermolinskiy, K. H. Kim, S. Shenker, I. Stoica, A data-oriented (and beyond) network architecture, in: SIGCOMM, 2007.
- o V. Ramasubramanian, R. Peterson, E. G. Sirer, Corona: a high performance publish-subscribe system for the world wide web, in: NSDI, 2006.
- o V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, R. L. Braynard, Networking Named Content, in: CoNEXT, 2009.
- o Y. Cui, B. Li, K. Nahrstedt, ostream: asynchronous streaming multicast in application-layer overlay networks, JSAC 22 (1) (2004) 91-106.
- o Y. Diao, S. Rizvi, M. J. Franklin, Towards an internet-scale XML dissemination service, in: VLDB, 2004.

And some related projects:

- o Named Data Networking (NDN) Project
- o Pursuit Project, <http://www.fp7-pursuit.eu/>
- o NetInf Project, <http://www.netinf.org>

6. Standardisation Considerations

Future versions of this document will outline a concrete protocol specification for pub/sub support for ICN. Below some initial standardisation considerations are outlined.

An initial list of details that need to be specified is the following:

- o Pub/Sub related interfaces/APIs
- o Pub/Sub related data structure modification to existing ICN proposals

We are also considering to write a survey paper that accumulates all the Pub/sub related work.

7. References

7.1. Normative References

- [RFC1112] Deering, S., "Host extensions for IP multicasting", STD 5, RFC 1112, August 1989.
- [RFC4601] Fenner, B., Handley, M., Holbrook, H., and I. Kouvelas, "Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)", RFC 4601, August 2006.

7.2. Informative References

- [Banerjee2002] Banerjee, S., Bhattacharjee, B., and C. Kommareddy, "Scalable application layer multicast", SIGCOMM, 2002, .
- [Carzaniga2011] Carzaniga, A., Papalini, M., and A. Wolf, "Content-based Publish/Subscribe Networking and Information-centric Networking", Proceedings of the ACM SIGCOMM workshop on Information-centric networking, ACM, 2011, .
- [Chen2011] Chen, J., Arumaithurai, M., Fu, X., and K. Ramakrishnan, "COPSS: An Efficient Content Oriented Publish/Subscribe System", ACM/IEEE 7th Symposium on Architectures for Networking and Communications Systems (ANCS), 2011, .
- [Chen2012] Chen, J., Arumaithurai, M., Fu, X., and K. Ramakrishnan, "G-COPSS: A Content Centric Communication Infrastructure for Gaming Applications", IEEE 32nd International Conference on Distributed Computing Systems (ICDCS), 2012, .
- [Chu2002] Chu, Y., Rao, S., Seshan, S., and H. Zhang, "A case for end system multicast", IEEE Journal on Selected Areas in Communications 20, no. 8 (2002): 1456-1471, .

[Fenner2005]

Fenner, W., Rabinovich, M., Ramakrishnan, K., Srivastava, D., and Y. Zhang, "XTreeNet: Scalable overlay networks for XML content dissemination and querying (synopsis)", 10th International Workshop on Web Content Caching and Distribution (WCW), 2005, .

[Jannotti2000]

Jannotti, J., Gifford, D., Johnson, K., and M. Kaashoek, "Overcast: reliable multicasting with on overlay network", Proceedings of the 4th conference on Symposium on Operating System Design & Implementation-Volume 4, pp. 14-14. USENIX Association, 2000, .

[Katsaros2011]

Katsaros, K., Xylomenos, G., and G. Polyzos, "MultiCache: An overlay architecture for information-centric networking", Computer Networks 55.4 (2011): 936-947, .

[Ramasubramanian2006]

Ramasubramanian, V., Peterson, R., and E. Sirer, "Corona: A High Performance Publish-Subscribe System for the World Wide Web", NSDI. Vol. 6. 2006, .

[Schmidt2012]

Schmidt, T. and M. Waehlisch, "Why We Shouldn't Forget Multicast in Name-oriented Publish/Subscribe", arXiv preprint arXiv:1201.0349 (2012), .

Appendix A. Acknowledgment

This document has been supported by the GreenICN project (GreenICN: Architecture and Applications of Green Information Centric Networking), a research project supported jointly by the European Commission under its 7th Framework Program (contract no. 608518) and the National Institute of Information and Communications Technology (NICT) in Japan (contract no. 167). The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the GreenICN project, the European Commission, or NICT.

Authors' Addresses

Mayutan Arumaithurai
University of Goettingen
Goldschmidt Str. 7
Goettingen 37077
Germany

Phone: +49 551 39 172046
Fax: +49 551 39 14416
Email: arumaithurai@informatik.uni-goettingen.de

Jiachen Chen
University of Goettingen
Goldschmidt Str. 7
Goettingen 37077
Germany

Phone: +49 551 39 172051
Fax: +49 551 39 14416
Email: jiachen@informatik.uni-goettingen.de

Xiaoming Fu
University of Goettingen
Goldschmidt Str. 7
Goettingen 37077
Germany

Phone: +49 551 39 172023
Fax: +49 551 39 14416
Email: fu@informatik.uni-goettingen.de

K. K. Ramakrishnan
University of California, Riverside
900 University Ave
Riverside CA 92521
USA

Email: kkramakrishnan@yahoo.com

Jan Seedorf
NEC
Kurfuerstenanlage 36
Heidelberg 69115
Germany

Phone: +49 6221 4342 221
Fax: +49 6221 4342 155
Email: seedorf@neclab.eu

ICN Research Group
Internet-Draft
Intended status: Experimental
Expires: January 5, 2015

A. Lindgren
F. Ben Abdesslem
SICS
O. Schelen
Lulea University of Technology
A. Malik
Ericsson
B. Ahlgren
SICS
July 4, 2014

Applicability and Tradeoffs of Information-Centric Networking for
Efficient IoT
draft-lindgren-icnrg-efficientiot-00

Abstract

This document outlines the tradeoffs involved in utilizing Information Centric Networking (ICN) for the Internet of Things (IoT) scenarios. It describes the contexts and applications where the IoT would benefit from ICN, and where a host-centric approach would be better. The requirements imposed by the heterogeneous nature of IoT networks are discussed (e.g., in terms of connectivity, power availability, computational and storage capacity). Design choices are then proposed for an IoT architecture to handle these requirements, while providing efficiency and scalability. An objective is to not require any IoT specific changes of the ICN architecture per se, but we do indicate some potential modifications of ICN that would improve efficiency and scalability for IoT and other applications.

This document mainly serves as a problem statement and will not present a conclusive architecture design. It can be used as a basis for further discussion and to design architectures for the IoT.

Status of this Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any

time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 5, 2015.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Motivation	4
2. Advantages of ICN Principles for IoT	5
2.1. Naming of Devices, Data and Services	5
2.2. Distributed Caching	5
2.3. Decoupling between Sender and Receiver	5
3. Design Challenges of IoT over ICN	7
3.1. Naming of Devices, Data and Services	7
3.2. Efficiency of Distributed Caching	8
3.3. Decoupling between Sender and Receiver	9
4. Proposed Design Choices for IoT over ICN	10
4.1. Existing Internet protocols	10
4.2. Data naming, format and composition	10
4.3. Immutable atomic data units	11
4.4. The importance of time	12
4.5. Decoupling and roles of senders and receivers	12
4.6. Combination of PULL/PUSH model	13
4.7. Capability advertisements	14
4.8. Name-based routing vs name resolution + 1-step vs 2-step	14
4.9. What's naming and what's searching	14
4.10. Tagging/tracing of data, and partial data	15
5. Other Issues	16
5.1. Security Considerations	16
5.1.1. Retrieving trusted content from several caches	16
5.1.2. Enabling application-layer processing in untrusted intermediaries	17
5.1.3. Energy efficiency of cryptographic mechanisms	17
6. Informative References	18
Authors' Addresses	19

1. Motivation

Information Centric Networking (ICN) has been shown to efficiently meet current usage demands of computer networks, where users consume content from the network instead of communicating with specific hosts. The applications and usage of the Internet of Things (IoT) often imply information centric usage patterns, where users or devices consume IoT generated content from the network instead of communicating with specific hosts or devices.

However, while the IoT shares many characteristics with typical information centric applications, it differs because of the high heterogeneity of connected devices, including mainly sensors and actuators, leading to different applications and usage. Because of these differences, applying an ICN approach to design the architecture of the IoT is often, but not always, beneficial. Depending on the context, the IoT architecture should follow an ICN approach, or a host-centric approach. In practice, the right approach is a complex tradeoff that depends on the applications and usage of the IoT network. This document describes some advantages and inconveniences of using an ICN architecture for the IoT, and helps finding the right tradeoff between an ICN and host-centric approach, depending on the context.

2. Advantages of ICN Principles for IoT

A key concept of ICN is the ability to name data independently from the current location at which it is stored, which simplifies caching and enables decoupling of sender and receiver. Using ICN concepts to design an architecture for IoT networks potentially provides these advantages compared to using traditional host-centric architecture. This section highlights general benefits an ICN architecture could provide to IoT networks in optimal contexts such as application's type, usage pattern, or network scale. Benefiting from the advantages described hereafter can only happen when taking into account the right tradeoff depending on the context, which will be discussed in the following section.

2.1. Naming of Devices, Data and Services

The heterogeneity of both network equipment deployed and services offered by IoT networks leads to a large variety of data, services and devices. While using a traditional host-centric architecture, only devices or their network interfaces are named at the network level, leaving to the application layer the task to name data and services. In many common applications of IoT networks, data and services are the main goal, and specific communication between two devices is secondary. The network distributes content and provides a service, instead of establishing a communication link between two devices. In this context, data content and services can be provided by several devices, or group of devices, hence naming data and services is often more important than naming the devices.

2.2. Distributed Caching

While caching mechanisms are already used by other types of overlay networks, IoT networks can potentially benefit even more from caching systems, because of their resource constraints. Wireless bandwidth and power supply can be limited for multiple devices sharing a communication channel, and for small mobile devices powered by batteries. In this case, avoiding unnecessary transmissions with IoT devices to retrieve and distribute IoT data to multiple places is important, and storing such content in the network can save wireless bandwidth and battery power. Moreover, as for other types of networks, applications for IoT networks requiring shorter delays can benefit from local caches to reduce delays between content request and delivery.

2.3. Decoupling between Sender and Receiver

IoT devices may be mobile and face intermittent network connectivity. When specific data is requested, such data can often be delivered by

ICN without any consistent direct connectivity between devices. Apart from using structured caching systems as described previously, information can also be spread by forwarding data opportunistically.

3. Design Challenges of IoT over ICN

As outlined in Section 2, there are potential benefits from using ICN to implement IoT communication architectures. However, in order to obtain a scalable and efficient architecture there are some aspects of ICN that must be specifically considered in making the right design choices for IoT. In fact, using an ICN approach may not be beneficial in all desired sub-functions and scenarios. This section outlines some of the ICN specific challenges that must be considered and describes some of the trade offs that will be involved. We will address these challenges in our proposed design choices later in Section 4.

3.1. Naming of Devices, Data and Services

Naming devices is a common element of both ICN and host-centric approaches. However, naming devices in the IoT raises different challenges that have to be addressed if an ICN approach is adopted. As for data and services, naming them in the network layer is proper to the ICN approach, and has to be designed carefully, depending on the context.

- o Naming of devices: Naming devices is often important when using an ICN approach in an IoT network. The presence of actuators requires clients to act specifically on a device, e.g. to switch it off. Also, managing and monitoring the devices for administration purposes requires devices to have a specific name allowing to identify them uniquely. There are multiple ways to achieve device naming, even in systems that are data centric by nature. For example, in systems that are adressable or searchable based on metadata or sensor content, the device identifier can be included as a special kind of metadata or sensor reading.
- o Size of data/service name: In information centric applications, the size of the data is often larger than its name. For the IoT, sensors and actuators are very common, and they can generate data as small as a short integer containing a temperature value, or a one-byte instruction to switch off an actuator. The name of the content for each of these pieces of data has to uniquely identify the content. For this reason, many existing naming schemes have long names that are likely to be longer than the actual data content for many types of IoT applications. Furthermore, naming schemes that have self certifying properties (e.g., by creating the name based on a hash of the content), suffer from the problem that the object can only be requested when the object has been created and the content is already known, thus requiring some form of indexing service. While this is an acceptable overhead for larger data objects, it is infeasible for use when the object size

is on the order of a few bytes.

- o Hash-based content name: Hash algorithms are commonly used to name content in order to verify that the content is the one requested. This is only possible in contexts where the requested object is already existing, and where there is a directory service to look up names. This approach is suitable for systems with large data objects where it is important to verify the content.
- o Metadata-based content name: Relying on metadata allows to generate a name for an object before it is created. However this mechanism requires metadata matching semantics.
- o Naming of services: Similarly to naming of devices or data, services can be referred to with a unique identifier, provided by a specific device or by someone assigned by a central authority as the service provider. It can however also be a service provided by anyone meeting some certain metadata conditions. Example of services include content retrieval, that takes a content name/description as input and returns the value of that content, and actuation, that takes an actuation command as input and possibly returns a status code afterwards.

3.2. Efficiency of Distributed Caching

Distributed caching is a key opportunity with ICN. However, an IoT framework must be carefully designed to reap the maximum benefits of ICN caching. When content popularity is heterogeneous, some content is often requested repeatedly. In that case, the network can benefit from caching. Another case where caching would be beneficial is when devices with low duty cycle are present in the network and when access to the cloud infrastructure is limited.

However, using distributed caching mechanisms in the network is not useful when each object is only requested at most once, as a cache hit can only occur for the second request and later. It may also be less useful and less scalable to have the caches distributed throughout the network in cases when all content is frequently requested. A better strategy in that case is to proactively send all data to central or distributed repositories (i.e., a central cache), possibly a cloud, from which all clients can retrieve the data, assuming the clients have good connectivity. Another example is when the name of the object has a different meaning depending on the context. For example, when the last value for a sensor reading is requested, the returned object will change every time the sensor reading is updated. In this case, caching cannot be used, and naming this as a service is more appropriate.

3.3. Decoupling between Sender and Receiver

Decoupling the sender and receiver is useful mechanism offered by the ICN approach, especially for content retrieval with duty cycling devices or devices with intermittent connectivity. However, in order to efficiently retrieve data it must be possible for requestors (receivers) to easily deduce the name of the data to request, without any direct contact with the responder (sender).

Nevertheless, this mechanism cannot be used when authentication is needed for management and actuation, or, of course, when real-time interaction between devices is necessary.

4. Proposed Design Choices for IoT over ICN

This section describes some fundamental design choices and trade-offs to allow for effective, efficient and scalable handling of IoT data in an ICN network. An objective with these choices is to facilitate that an ICN network can be used without requiring additions of IoT application specific functionality in the ICN network. However, in some cases we do invite for discussion on tentative additions of functionality to ICN in order to make the overall IoT solution more efficient and scalable.

4.1. Existing Internet protocols

IoT devices can have a role as content generators (e.g., sensors) in where an ICN paradigm should be effective for data retrieval and dissemination. However, IoT devices may also have roles as actuators in which such devices shall be accessed for control purposes. The use of an ICN network may be less natural when actuation and control of specific devices is the key objective. To facilitate support of IoT for both data generation and control/actuation, we assume that there is a need for existing internet protocols, and the ICN routing should therefore work in concert with existing Internet protocols.

4.2. Data naming, format and composition

The data served by ICN may be aggregated from smaller components. Although IoT data components in many cases are small and simple, a general challenge in defining ICN applications is to decide how to compose (i.e. group) the data so that it can be effectively named and requested. Requesting partial data inside a composition may become a challenge. Indeed, if data is composed and sub components are requested, which are not directly namable by the requestor, finding such a subset will resemble a database query which may require processing to resolve. The ICN network should not have to support such complexity.

A design choice regarding IoT data is therefore to keep the ICN network free from supporting any advanced queries and instead only support directly addressable (i.e., named) data units. Any advanced composition (hierarchical, graph-based, hyperlink, etc.) of IoT data, and related searching for sub-components, would be handled in servers/endpoints instead of inside the ICN network. The issue of structure and searching is for further study. For effective ICN interoperability, only the structure of the atomic addressable data units must be agreed. There are several advantages of this design choice. First, the size of the directly addressable units can be kept fairly small to avoid that unwanted bulk data is pulled over resource constrained networks or spread over various caches in the

ICN network. This results in better resource utilization, better localization of desired data, and ultimately better scalability. There is however one tradeoff in that smaller data units results in a larger overhead. Second, the computational requirement is kept low in the ICN network, essentially limiting it to deciding whether there is a cache hit or not. Third, few new requirements are put on ICN data dissemination. Existing methods will be sufficient. Fourth, this simplification means that a flat address space would be sufficient, but for practical reasons a hierarchical address space may be preferred. There is flexibility in the choice of exact addressing scheme and it may depend on which existing ICN framework that is used for IoT data.

4.3. Immutable atomic data units

The number of IoT devices as well as the amount of data produced by these devices may potentially be very large, and the data may be spread over very large ICN networks. The potential problem of cache inconsistencies in an ICN network may therefore be large if we allow for data to be mutable objects. To support scalability and horizontal distribution it is essential to define data properties that facilitate independency and consistency, while minimizing the need for dynamic global synchronization.

A key design choice is therefore to mandate that IoT only uses immutable atomic data units. This supports large scale distribution by ensuring that there is no stale data in the ICN domain. A hit is always a clean hit. A trade-off from this is that dynamic data must be modeled as a stream of immutable data units, potentially consuming more resources. However, this challenge can be resolved by smart caching strategies where old data is dropped. A client that wants the "latest" reading can according to our previously mentioned design choice, in Section 4.2, not ask the ICN network such a high level query, instead it must ask for the specific (version of) information. There are several methods for finding the latest version, for example through a high level request from a server/endpoint, or by using a naming scheme where the name can be directly inferred, e.g., if an IoT device has advertised that it produces data every whole second, the named data can include absolute time and therefore data from the current second can be requested (provided that clock synchronisation is accurate enough, which is out of scope of this document). These methods are based on the request/pull method. For real-time update (most accurate info), there is also an option to use dissemination based on the push model as described later in Section 4.6.

4.4. The importance of time

In Section 4.3 we started to discuss the role of time in relation to immutable data. We want to emphasize that time almost always is a very important property of IoT data, and especially so for data that change over time. When modeling dynamic IoT data with a stream of immutable data, it is often the case that a certain IoT data object is a sensor reading at a particular point in time, and the next object in the stream is the next reading. Thus, dynamic data is in this case dynamic over time, with well defined (immutable) values for particular points in time.

We therefore argue that it is important to find a way to represent these time-related streams of immutable data. It should be possible to request data from a certain time, and to infer/find the name of the latest, most current, data. As mentioned in Section 4.3, an IoT device might advertise that it produces data at certain time intervals. This information is also useful for the ICN network to be able to handle requests for the corresponding data in the most efficient manner.

It is for further study whether any extensions are needed to the ICN paradigm, or if it can be supported with, e.g. clever use of metadata, namespace, and search functionality. It may also depend on the particular flavor of ICN. The naming scheme of CCN/NDN may here provide an advantage.

We also note that time is also important for other applications, in particular for live streaming video. Live video also produces a time-related stream of immutable objects, and would in the same way benefit from such support in the ICN service.

4.5. Decoupling and roles of senders and receivers

Since ICN networks essentially support a request/response model of interaction, we denote the receivers of information as requestors, and the senders of information as responders. The ICN network in itself provides decoupling of requestors and responders, but it does not (and should not) provide any transformation or aggregation of data. The IoT dissemination architecture should therefore allow for any number of intermediate processing nodes. An intermediate node will be an endpoint in the ICN network that can act as both requestor and responder. Such a node may perform aggregation, filtering, selection, etc. The instantiation of such nodes may for example form a directed (acyclic) graph between ultimate responders (IoT devices) and ultimate requestors (the final applications). It is for further study how to define such an architecture.

It is a design choice to keep the IoT dissemination and aggregation functionality outside of the ICN domain. That architecture would be an overlay that may have intricate structure, and put the ICN usage in a new context, where content from ultimate requestors to ultimate responders may go through many IoT processing nodes that collect, process and re-publish data through an ICN for various purposes.

4.6. Combination of PULL/PUSH model

A critical decision regarding IoT data is whether to use a PULL model, a PUSH model, or both. There are some intrinsic trade offs between these models. The PULL model is for example resource efficient when there is an abundant amount of IoT information, potentially redundant from many devices, and the clients only occasionally or partially are interested in the information. The PUSH model is for example efficient when there is real-time information and the clients are interested in all information from specific devices all the time.

A design decision in the IoT domain is to support both PULL and PUSH. The base model should be PULL, meaning that requestors must always start by sending a request. If the request is for some specific data, it can be resolved by returning the data (if it exists). The pull model can be supported efficiently and scalably by an ICN network. A request can however also include triggers, which means that data will be returned (pushed) when triggers are fulfilled, which may be immediately, or in the future at one or several occasions. This can be used to select alarm conditions, to request continuous or periodic push, etc. The trigger conditions can be set by the requestor, or be pre-defined by the responder. The former is more flexible but may have performance/scalability issues. The latter is more scalable since there will be a predefined and finite number of trigger conditions. Our recommended choice, at least for the initial phase, is to go for a simple and scalable solution and therefore adopt the model where available trigger conditions are defined and advertised by the responder. The ICN would be apt for supporting such capability advertisements, given that they are fairly static.

We recommend to have a discussion on whether an ICN network can or should provide an option to effectively support a push model of data. Such support can make real-time IoT data dissemination more efficient and scalable as previously mentioned in Section 4.3. However, since we assume that the ICN works with existing IP protocols, such functionality can be provided without ICN, by using traditional unicast or multicast communication. We finally note that an ICN supported push service model would make the ICN network more like a publish/subscribe system.

4.7. Capability advertisements

Capability advertisements and discovery can be used by requestors to discover which responders to connect to. In a deployment with large numbers of responders, the functionality of automatic advertisement and discovery becomes a critical factor to support scaling. Responders should advertise their methods (inputs, outputs, parameters, triggers, etc) and provide relevant metadata. Such capability advertisements should be conservative with resources, which suggests that new advertisements should be posted with reasonably low frequency. This implies that an ICN network can be used for providing capability advertisements. The advertisements should be provided as a stream of immutable objects, or alternatively the system should be tolerant to stale caches. Should there be a need real-time awareness of dynamic changes, a push model of capability advertisements could be used as earlier described in Section 4.6.

4.8. Name-based routing vs name resolution + 1-step vs 2-step

As described in Section 4.2, the IoT framework should be defined so that new functionality in the ICN is not needed. For data that is frequently generated and regenerated, it makes sense to keep simple structures and provide directly inferable naming/addressing of data objects, so that requestors can directly address the data. For more complex data, such as pre-processed, aggregated and structured data a two-step resolution model is recommended. The IoT devices can provide a higher level resolution based on for example queries and searching, resulting in a number of concrete directly addressable ICN objects. This is similar to what web servers do when they return URLs that requestors can use, but in this case it is named content that is returned.

Consequently, the IoT framework should have no requirement that the ICN network itself should support 2-step addressing (although such 2-step methods may exist in some ICNs)

4.9. What's naming and what's searching

As described in Section 4.2, the IoT framework should be defined so that no new functionality is required in the ICN for searching data or subcomponents of data. The ICN network supports just naming of atomic data objects, while any searching is provided by the IoT framework, which in itself may be constituted by a highly distributed set of nodes that provide processing, analysis and aggregation of IoT data.

4.10. Tagging/tracing of data, and partial data

IoT data may be tagged with metadata to tell where it originates from. Tagging is made at the level above the ICN network and may for example be a list of strings. It can be added/changed by the originating node (or a node that assigns the originating ID), and added/changed/deleted by any node that processes the data. The tag can in some cases be used to trace data back to origins. For the ICN network, the metadata units are just black-box data that is to be conveyed, and therefore are not to modify the tags. However, in some cases it makes no sense to transmit any metadata. For efficiency reasons the ICN network should have support for optional delivery of metadata. This is to be conservative with scarce resources, for example when a wireless node requests data which is cached in the ICN network, it would be beneficial if the requestor could tell that it is desirable to not receive any metadata. There should be a discussion whether there should be just one, or more than one, piece of optional information in ICN content to be future proof.

5. Other Issues

5.1. Security Considerations

The ICN paradigm is content-centric as opposed to state-of-the-art host-centric internet. Besides aspects like naming, content retrieval and caching this also has security implications. ICN advocates the model of trust in content rather than trust in network hosts. This brings in the concept of Object Security which is contrary to session-based security mechanisms such as TLS/DTLS prevalent in the current host-centric internet.

Object Security is based on the idea of securing information objects unlike session-based security mechanisms which secure the communication channel between a pair of nodes. This reinforces an inherent characteristic of ICN networks i.e. to decouple senders and receivers. In the context of IoT, the Object Security model has several concrete advantages. As discussed earlier in Section 2.1, in many IoT applications data and services are the main goal and specific communication between two devices is secondary. Therefore it makes more sense to secure IoT objects instead of securing the session between communicating endpoints.

It is important that while security mechanisms complement the ICN architecture in a coherent fashion, they do so without laying down any strict requirements or constraints. Therefore, the decision of what security mechanisms are employed should be handled at a layer above ICN, in this case within the IoT framework. This facilitates flexibility and allows IoT applications more freedom to decide what security mechanism suits them best (session-based security, object security or a hybrid). Though the idea of Object Security is very much inline with the ICN concept, there can still be some use cases where Object Security does not add much e.g. a Pub/Sub interaction where a client is expected to interact more or less with the same server node (a session-based security protocol should suffice here) or use cases where application layer headers should also be secured (which can be achieved by TLS/DTLS). We, therefore, effectively imply that there is no need to modify typical ICN standards to accommodate Object Security.

The following sub-sections discuss some advantages of using Object Security in IoT applications.

5.1.1. Retrieving trusted content from several caches

When functioning in an ICN network, an IoT client is expected to rely on the network to deliver the requested content in an optimal fashion without concerning itself with where the content actually lies. This

could potentially mean that each individual object within a stream of immutable objects is retrieved from a different source. Having a trust relationship with each of these different sources is not realistic. This gives rise to the need of retrieving trusted content from untrusted nodes/caches in an ICN network. Object security is ideal in such use cases because it relieves an IoT client application from the hassle of having to establish trust with each node that can potentially cache an IoT object. This also means that a requesting client can make use of more caches in the network, hence resulting in better throughput and latency.

5.1.2. Enabling application-layer processing in untrusted intermediaries

Object Security ensures that objects in application-layer payload are secure e.g. XML, JSON objects. However, the application-layer header is unencrypted and available for processing. Securing content at the object level means greater granularity. This facilitates application-layer processing in untrusted intermediary nodes (e.g. proxies and caches) without compromising security. An example use case is untrusted caching nodes that should have the ability to cache individual encrypted objects without being able to see what is there in those objects. In this case there is a need for the caching nodes to identify the object URI which can be done by looking into the application-layer header. But the object is still encrypted and unknown to the caching nodes.

5.1.3. Energy efficiency of cryptographic mechanisms

Session-based security protocols rely on the exchange of several messages before a secure session is established between a pair of nodes. Use of such protocols in constrained IoT devices can have serious consequences in terms of power efficiency because in most cases transmission and reception of messages is more costly than the cryptographic operations. This is especially true for wireless devices. The problem is amplified even further when the constrained device is interacting with a number of caching nodes because the device will have to setup a secure session with each caching node. The Object Security model eliminates this problem because the content is readily available in a secure state in the network. IoT devices producing data can secure it w.r.t. all the intended consumers and start transmitting it right away.

6. Informative References

[vahdat_00]

Vahdat, A. and D. Becker, "Epidemic Routing for Partially Connected Ad Hoc Networks", Duke University Technical Report CS-200006, April 2000.

Authors' Addresses

Anders F. Lindgren
SICS Swedish ICT
Box 1263
Kista SE-164 29
SE

Phone: +46707177269
Email: andersl@sics.se
URI: <http://www.sics.se/~andersl>

Fehmi Ben Abdesslem
SICS Swedish ICT
Box 1263
Kista SE-164 29
SE

Phone: +46705470642
Email: fehmi@sics.se
URI: <http://www.sics.se/~fehmi>

Olov Schelen
Lulea University of Technology
Lulea SE-971 87
SE

Phone:
Email: olov.schelen@ltu.se
URI:

Adeel Mohammad Malik
Ericsson
Kista SE-164 80
SE

Phone: +46725074492
Email: adeel.mohammad.malik@ericsson.com
URI:

Bengt Ahlgren
SICS Swedish ICT
Box 1263
Kista SE-164 29
SE

Phone: +46703141562
Email: bengta@sics.se
URI: <http://www.sics.se/people/bengt-ahlgren>

ICNRG
Internet-Draft
Intended status: Informational
Expires: December 29, 2014

J. Seedorf
NEC
M. Arumaithurai
University of Goettingen
A. Tagami
KDDI R&D Labs
K. Ramakrishnan
University of California
N. Blefari Melazzi
University Tor Vergata
June 27, 2014

Using ICN in disaster scenarios
draft-seedorf-icn-disaster-02

Abstract

Information Centric Networking is a new paradigm where the network provides users with named content, instead of communication channels between hosts. This document outlines some research directions for Information Centric Networking (ICN) with respect to applying ICN approaches for coping with natural or human-generated, large-scale disasters.

Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on December 29, 2014.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. Disaster Scenarios	3
3. Research Challenges and Benefits of ICN	4
3.1. High-Level Research Challenges	4
3.2. How ICN can be Beneficial	5
4. Use Cases and Requirements	6
5. Solution Design	7
6. The GreenICN Project	8
7. Conclusion	9
8. References	9
8.1. Normative References	9
8.2. Informative References	10
Appendix A. Acknowledgment	10
Authors' Addresses	10

1. Introduction

This document summarizes some research challenges for coping with natural or human-generated, large-scale disasters. Further, the document discusses potential directions for applying Information Centric Networking (ICN) to address these challenges.

Section 2 gives some examples of what can be considered a large-scale disaster and what the effects of such disasters on communication networks are. Section 3 outlines why ICN can be beneficial in such scenarios and provides a high-level overview on corresponding research challenges. Section 4 describes some concrete use cases and requirements for disaster scenarios. In Section 5, some concrete ICN-based solutions approaches are outlined. Related research activities are ongoing in the GreenICN research project; Section 6 provides an overview of this project.

2. Disaster Scenarios

An enormous earthquake hit Northeastern Japan (Tohoku areas) on March 11, 2011, and caused extensive damages including blackouts, fires, tsunamis and a nuclear crisis. The lack of information and means of communication caused the isolation of several Japanese cities. This impacted the safety and well-being of residents, and affected rescue work, evacuation activities, and the supply chain for food and other essential items. Even in the Tokyo area that is 300km away from the Tohoku area, more than 100,000 people became 'returner' refugees, who could not reach their homes because they had no means of public transportation (the Japanese government has estimated that more than 6.5 million people would become returner refugees if such a catastrophic disaster were to hit the Tokyo area).

That earthquake in Japan also showed that the current network is vulnerable against disasters and that mobile phones have become the lifelines for communication including safety confirmation. The aftermath of a disaster puts a high strain on available resources due to the need for communication by everyone. Authorities such as the President/Prime-Minister, local authorities, Police, fire brigades, and rescue and medical personnel would like to inform the citizens of possible shelters, food, or even of impending danger. Relatives would like to communicate with each other and be informed about their wellbeing. Affected citizens would like to make enquiries of food distribution centres, shelters or report trapped, missing people to the authorities. Moreover, damage to communication equipment, in addition to the already existing heavy demand for communication highlights the issue of fault-tolerance and energy efficiency.

Additionally, disasters caused by humans such as a terrorist attack may need to be considered, i.e. disasters that are caused deliberately and willfully and have the element of human intent. In such cases, the perpetrators could be actively harming the network by launching a Denial-of-Service attack or by monitoring the network passively to obtain information exchanged, even after the main disaster itself has taken place. Unlike some natural disasters that are predictable using weather forecasting technologies and have a slower onset and occur in known geographical regions and seasons, terrorist attacks may occur suddenly without any advance warning. Nevertheless, there exist many commonalities between natural and human-induced disasters, particularly relating to response and recovery, communication, search and rescue, and coordination of volunteers.

The timely dissemination of information generated and requested by all the affected parties during and the immediate aftermath of a disaster is difficult to provide within the current context of global

information aggregators (such as Google, Yahoo, Bing etc.) that need to index the vast amounts of specialized information related to the disaster. Specialized coverage of the situation and timely dissemination are key to successfully managing disaster situations. We believe that network infrastructure capability provided by Information Centric Networks can be suitable, in conjunction with application and middleware assistance.

3. Research Challenges and Benefits of ICN

3.1. High-Level Research Challenges

Given a disaster scenario as described in Section 2, on a high-level one can derive the following (incomplete) list of corresponding technical challenges:

- o Enabling usage of functional parts of the infrastructure, even when these are disconnected from the rest of the network: Assuming that parts of the network infrastructure (i.e. cables/links, routers, mobile bases stations, ...) are functional after a disaster has taken place, it is desirable to be able to continue using such components for communication as much as possible. This is challenging when these components are disconnected from the backhaul, thus forming fragmented networks. This is especially true for today's mobile networks which are comprised of a centralised architecture, mandating connectivity to central entities (which are located in the core of the mobile network) for communication. But also in fixed networks, access to a name resolution service is often necessary to access some given content.
- o Decentralised authentication: In mobile networks, users are authenticated via central entities. In order to communicate in fragmented or disconnected parts of a mobile network, the challenge of decentralising such user authentication arises. Independently of the network being fixed or mobile, data origin authentication of content retrieved from the network is challenging when being 'offline' (e.g. disconnected from servers of a security infrastructure such as a PKI).
- o Delivering/obtaining information in congested networks: Due to broken cables, failed routers, etc., it is likely that in a disaster scenario the communication network has much less overall capacity for handling traffic. Thus, significant congestion can be expected in parts of the infrastructure. It is therefore a challenge to guarantee message delivery in such a scenario. This is even more important as in the case of a disaster aftermath, it

may be crucial to deliver certain information to recipients (e.g. warnings to citizens).

- o Delay/Disruption Tolerant Approach: Fragmented networks makes it difficult to support end-to-end communication. However, communication in general and especially during disaster can tolerate some form of delay. E.g. in order to know if his/her relatives are safe or a 'SOS' call need not be supported in an end-to-end manner. It is sufficient to improve communication resilience in order to deliver such important messages.
- o Energy Efficiency: Long-lasting power outages may lead to batteries of communication devices running out, so designing energy-efficient solutions is very important in order to maintain a usable communication infrastructure.

The list above is most likely incomplete; future revisions of this document intend to add additional challenges to the list.

3.2. How ICN can be Beneficial

Several aspects of ICN make related approaches attractive candidates for addressing the challenges described in Section 3.1. Below is an (incomplete) list of considerations why ICN approaches can be beneficial to address these challenges:

- o Routing-by-name: ICN protocols natively route by named data objects and can identify objects by names, effectively moving the process of name resolution from the application layer to the network layer. This functionality is very handy in a fragmented network where reference to location-based, fixed addresses may not work as a consequence of disruptions. For instance, name resolution with ICN does not necessarily rely on the reachability of application-layer servers (e.g. DNS resolvers). In highly decentralised scenarios (e.g. in infrastructureless, opportunistic environments) the ICN routing-by-name paradigm effectively may lead to a 'replication-by-name' approach, where content is replicated depending on its name.
- o Authentication of named data objects: ICN is built around the concept of named data objects. Several proposals exist for integrating the concept of 'self-certifying data' into a naming scheme (see e.g. [RFC6920]). With such approaches, the origin of data retrieved from the network can be authenticated without relying on a trusted third party or PKI.
- o Content-based access control: ICN can regulate access to data objects (e.g. only to a specific user or class of users) by means

of content-based security; this functionality could facilitate trusted communications among peer users in isolated areas of the network.

- o Caching: Caching content along a delivery path is an inherent concept in ICN. Caching helps in handling huge amounts of traffic, and can help to avoid congestion in the network (e.g. congestion in backhaul links can be avoided by delivering content from caches at access nodes).
- o Sessionless: ICN does not require full end-to-end connectivity. This feature facilitates a seamless aggregation between a normal network and a fragmented network, which needs DTN-like message forwarding.

The list above is most likely incomplete; future revisions of this document intend to add more considerations to the list and to argue in more detail why ICN is suitable for addressing the aforementioned research challenges.

4. Use Cases and Requirements

This Section describes some use cases for the aforementioned disaster scenario (as outlined in Section 2) and discusses the corresponding technical requirements for enabling these use cases.

- o Delivering Messages to Relatives/Friends: After a disaster strikes, citizens want to confirm to each other that they are safe. For instance, shortly after a large disaster (e.g., Earthquake, Tornado), people have moved to different refugee shelters. The mobile network is not fully recovered and is fragmented, but some base stations are functional. This use case imposes the following high-level requirements: a) People must be able to communicate with others in the same network fragment, b) people must be able to communicate with others that are located in different fragmented parts of the overall network. More concretely, the following requirements are needed to enable the use case: a) a mechanism for scalable message forwarding scheme that dynamically adapts to changing conditions in disconnected networks, b) DTN-like mechanisms for getting information from disconnected island to another disconnected island, and c) data origin authentication so that users can confirm that the messages they receive are indeed from their relatives or friends.
- o Spreading Crucial Information to Citizens: State authorities want to be able to convey important information (e.g. warnings, or information on where to go or how to behave) to citizens. These kinds of information shall reach as many citizens as possible.

i.e. Crucial content from legal authorities shall potentially reach all users in time. The technical requirements that can be derived from this use case are: a) Data origin authentication, such that citizens can confirm the authenticity of messages sent by authorities, b) mechanisms that guarantee the timeliness and loss-free delivery of such information, which may include techniques for prioritizing certain messages in the network depending on who sent them, and c) DTN-like mechanisms for getting information from disconnected island to another disconnected island.

It can be observed that different key use cases for disaster scenarios imply overlapping and similar technical requirements for fulfilling them. As discussed in Section 3.2, ICN approaches are envisioned to be very suitable for addressing these requirements with actual technical solutions.

5. Solution Design

This Section outlines some ICN-based approaches that aim at fulfilling the previously mentioned use cases and requirements.

- o ICN 'data mules': To facilitate the exchange of messages between different network fragments, mobile entities can act as ICN 'data mules' which are equipped with storage space and move around the disaster-stricken area gathering information to be disseminated. As the mules move around, they deliver messages to other individuals or points of attachment to different fragments of the network. These 'data mules' could have a pre-determined path (an ambulance going to and from a hospital), a fixed path (drone/robot assigned specifically to do so) or a completely random path (doctors moving from one camp to another).
- o Priority dependent Name-based replication: By allowing spatial and temporal scoping of named messages, priority based replication depending on the scope of a given message is possible. Clearly, spreading information in disaster cases involves space and time factors that have to be taken into account as messages spread. A concrete approach for such scope-based prioritisation of ICN messages in disasters, called 'NREP', has been proposed [Psaras2014], where ICN messages have attributes such as user-defined priority, space, and temporal-validity. These attributes are then taken into account when prioritizing messages. In [Psaras2014], evaluations show how this approach can be applied to the use case 'Delivering Messages to Relatives/Friends' described in Section 4

- o Data-centric confidentiality and access control: In ICN, the requested content is not anymore associated to a trusted server or an endpoint location, but it can be retrieved from any network cache or a replica server. This call for 'data-centric' security, where security relies on information exclusively contained in the message itself, or, if extra information provided by trusted entities is needed, this should be gathered through offline, asynchronous, and non interactive communication, rather than from an explicit online interactive handshake with trusted servers. The ability to guarantee security without any online entities is particularly important in disaster scenarios with fragmented networks. One concrete cryptographic technique is 'Ciphertext-Policy Attribute Based Encryption' (CP-ABE), allowing a party to encrypt a content specifying a policy, which consists in a Boolean expression over attributes, that must be satisfied by those who want to decrypt such content. Such encryption schemes tie confidentiality and access-control to the transferred data, which can be transmitted also in an unsecured channel, enabling the source to specify the set of nodes allowed to decrypt.
- o Decentralised authentication of messages: Self-certifying names provide the property that any entity in a distributed system can verify the binding between a corresponding public key and the self-certifying name without relying on a trusted third party. Self-certifying names thus provide a decentralized form of data origin authentication. However, self-certifying names lack a binding with a corresponding real-world identity. Given the decentralised nature of a disaster scenario, a PKI-based approach for binding self-certifying names with real-world identities is not feasible. Instead, a Web-of-Trust can be used to provide this binding. Not only are the cryptographic signatures used within a Web-of-Trust independent of any central authority; there are also technical means for making the inherent trust relationships of a Web-of-Trust available to network entities in a decentralised, 'offline' fashion, such that information received can be assessed based on these trust relationships. A concrete scheme for such an approach has been published in [Seedorf2014], where also concrete examples for fulfilling the use case 'Delivering Messages to Relatives/Friends' with this approach are given.

6. The GreenICN Project

This section provides a brief overview of the GreenICN project. You can find more information at the project web site <http://www.greenicn.org/>

The recently formed GreenICN project, funded by the EU and Japan, aims to accelerate the practical deployment of ICN, addressing how

ICN networks and devices can operate in a highly scalable and energy-efficient way. The project will exploit the designed infrastructure to support multiple applications including the following two broad exemplary scenarios: 1) The aftermath of a disaster, e.g. hurricane, earthquake, tsunami, or a human-generated network breakdown when energy and communication resources are at a premium and it is critical to efficiently distribute disaster notification and critical rescue information. Key to this is the ability to exploit fragmented networks with only intermittent connectivity, the potential exploitation of multiple modalities of communication and use of query/response and pub/sub approaches; 2) Scalable, efficient pub/sub video delivery, a key requirement in both normal and disaster situations.

GreenICN will expose a functionality-rich API to spur the creation of new applications and services expected to drive industry and consumers, with special focus on the EU and Japanese environments, into ICN adoption. Our team, comprising researchers with diverse expertise, system and network equipment manufacturers, device vendors, a startup, and mobile telecommunications operators, is very well positioned to design, prototype and deploy GreenICN technology, and validate usability and performance of real-world GreenICN applications, contributing to create a new, low-energy, Information-Centric global communications infrastructure. We also plan to make contributions to standards bodies to further the adoption of ICN technologies.

7. Conclusion

This document outlines some research directions for Information Centric Networking (ICN) with respect to applying ICN approaches for coping with natural or human-generated, large-scale disasters. The document describes high-level research challenges as well as a general rationale why ICN approaches could be beneficial to address these challenges. One main objective of this document is to gather feedback from the ICN community within the IETF and IRTF regarding how ICN approaches can be suitable to solve the presented research challenges. Future revisions of this draft intend to include additional research challenges and to discuss what implications this research area has regarding related, future IETF standardisation.

8. References

8.1. Normative References

- [RFC6920] Farrell, S., Kutscher, D., Dannewitz, C., Ohlman, B., Keranen, A., and P. Hallam-Baker, "Naming Things with Hashes", RFC 6920, April 2013.

8.2. Informative References

[Psaras2014]

Psaras, I., Saino, L., Arumaithurai, M., Ramakrishnan, K., and G. Pavlou, "Name-Based Replication Priorities in Disaster Cases", 2nd Workshop on Name Oriented Mobility (NOM), 2014, .

[Seedorf2014]

Seedorf, J., Kutscher, D., and F. Schneider, "Decentralised Binding of Self-Certifying Names to Real-World Identities for Assessment of Third-Party Messages in Fragmented Mobile Networks", 2nd Workshop on Name Oriented Mobility (NOM), 2014, .

Appendix A. Acknowledgment

The authors would like to thank Ioannis Psaras for useful comments.

This document has been supported by the GreenICN project (GreenICN: Architecture and Applications of Green Information Centric Networking), a research project supported jointly by the European Commission under its 7th Framework Program (contract no. 608518) and the National Institute of Information and Communications Technology (NICT) in Japan (contract no. 167). The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the GreenICN project, the European Commission, or NICT.

Authors' Addresses

Jan Seedorf
NEC
Kurfuerstenanlage 36
Heidelberg 69115
Germany

Phone: +49 6221 4342 221
Fax: +49 6221 4342 155
Email: seedorf@neclab.eu

Mayutan Arumaithurai
University of Goettingen
Goldschmidt Str. 7
Goettingen 37077
Germany

Phone: +49 551 39 172046
Fax: +49 551 39 14416
Email: arumaithurai@informatik.uni-goettingen.de

Atsushi Tagami
KDDI R&D Labs
2-1-15 Ohara
Fujimino, Saitama 356-85025
Japan

Phone: +81 49 278 73651
Fax: +81 49 278 7510
Email: tagami@kddilabs.jp

K. K. Ramakrishnan
University of California
Riverside CA
USA

Email: kkramakrishnan@yahoo.com

Nicola Blefari Melazzi
University Tor Vergata
Via del Politecnico, 1
Roma 00133
Italy

Phone: +39 06 7259 7501
Fax: +39 06 7259 7435
Email: blefari@uniroma2.it

ICNRG
Internet-Draft
Intended status: Informational
Expires: December 27, 2014

J. Seedorf
NEC
June 25, 2014

Binding Self-certifying Names to Real-World Identities with a Web-of-Trust
draft-seedorf-icn-wot-selfcertifying-00

Abstract

Self-certifying names are one way of binding a given public key to a certain name in Information Centric Networking. However, an additional binding of a self-certifying name to a Real-World identity is needed in most cases, so that a recipient of some information cannot only verify that the publisher was in possession of the correct corresponding private key for the requested name, but that in addition the name itself is the intended one. This draft specifies how such a binding of Real-World identities with self-certifying ICN names can be done, taking existing IETF specifications into account.

Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on December 27, 2014.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of

publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. High-Level Design	3
3. Standardisation Considerations	4
4. Conclusion	4
5. References	5
5.1. Normative References	5
5.2. Informative References	5
Appendix A. Acknowledgment	6
Author's Address	6

1. Introduction

Self-certifying names provide the useful property that any entity in a distributed system can verify the binding between a corresponding public key and the self-certifying name without relying on a trusted third party [Aura2003]. Self-certifying names thus provide a decentralized form of data origin authentication. This feature makes self-certifying names a prime candidate for addressing the security requirements in Information Centric Networking (ICN) (which are inherently different from IP networks): a source can digitally sign data associated with a self-certifying name, and any intermediate entity (e.g. ICN-router/Cache) or receiving entity (i.e. issuer of a request for the name) can verify the signature, without the need to verify the identity of the host that caches the object, nor relying on a trusted third party, or a Public Key Infrastructure (PKI). However, as noted in [Ghods2011] and elsewhere, self-certifying names lack a binding with a corresponding real-world identity (RWI): the concept enables to verify that whoever signed some data was in possession of the private key associated with the self-certifying name, but it does not provide any means to verify what real-world identity corresponds to the public key, i.e. who actually signed the data [Ghods2011] [Nom2014].

In principle, this binding between a public key and an RWI could be provided by a PKI, or alternatively by a Web-of-Trust (WoT) [Ghods2011]. Several ICN approaches use a PKI [Survey]. However, until recently, there have not been concrete proposals for a WoT-based approach for binding a public key (or a self-certifying name) with an RWI in content-oriented architectures. A concrete approach

on how this can be done has been proposed in [Nom2014]. This document has the objective of providing the corresponding necessary standards specification to enable this approach (or similar ones) in principle in an interoperable way.

2. High-Level Design

On a high level, binding of self-certifying names and a Web-of-Trust can be achieved in the following way (see [Nom2014] for a detailed example of such an approach): The WoT key-ID is equivalent to the self-certifying name part used in the naming scheme. This ties the self-certifying name with the ID of the corresponding public key in the WoT.

For instance, in the existing PGP Web-of-Trust, the V4 key ID is the lower 64 bits of the fingerprint of the public key, where the fingerprint is essentially the 160-bit SHA-1 hash of the public key [RFC2440]. So if a self-certifying name would be based on the same lower 64-bits of the fingerprint of a given public key, this public key would be tied to the self-certifying name and at the same time be tied to the real-world identity used in the WoT, e.g. an email-address or the real (i.e. non-self-certifying) name of a given ICN publisher.

Thus, if a user requests the content for a self-certifying name in a given ICN architecture, he/she would retrieve the content which contains a digital signature and the corresponding public key for the self-certifying name. The user can then verify that the content retrieved indeed belongs to the name by first hashing the public key and confirm that the hash (or part of it) matches the requested name, and second using the public key to verify the signature over the content. This is in principle the general way of using self-certifying names for data origin authentication in distributed systems. If, in addition, (part of) the self-certifying name is equivalent to a WoT key-ID, the user can use any WoT infrastructure (e.g. PGP keyservers) to retrieve certificates for the key ID that contain/confirm the binding between the corresponding (to the WoT key ID) public key with a real-world identity, such as an email address. This binding provides the requesting user with assurance that the self-certifying name indeed is owned by the intended publisher, i.e. is the correct, intended name from the requestor's perspective.

The current PGP specification [RFC2440] considers only a bitlength of 64-bit for forming the key-ID, which is not very collision-resistant (collision-resistance among different key-IDs was not a design goal for PGP [RFC2440]). For securely binding a self-certifying name to a WoT key-ID, collision-resistance is a design goal, because otherwise attackers could potentially forge a binding of their public key with

a given self-certifying name. Thus, either a longer bitlength of the hash of the public key (or its fingerprint) must be used, or hash extension techniques [Aura] must be used, which effectively make collision attacks harder for constant bitlengths at the price of the time needed to create a public/private key pair. Future versions of this document will take these design considerations into account.

3. Standardisation Considerations

Future versions of this document will outline a concrete protocol specification for binding self-certifying names to a Web-of-Trust as outlined on a high level in the previous Section. Below some initial standardisation considerations are highlighted. Also, future versions of this document will look in more detail into existing IETF specifications, e.g. regarding ICN naming ([RFC6920]) and Web-of-Trust ([RFC2440]), and inspect to what extent such existing specifications can be used directly or in a modified form.

An initial list of details that need to be specified is the following:

- o (List of) Asymmetric cryptography algorithm(s) and corresponding bit-length(s)
- o (List of) Hash algorithm(s) and corresponding bit-length(s)
- o Rules that define what part of the hash is used for forming the self-certifying part of the name
- o Rules for forming a self-certifying name based on a public key
- o Semantics of a signature in the Web-of-Trust
- o Definition of the web-of-trust key-ID and how it relates to the self-certifying name
- o Definition of how many bits are used in case of hash extension techniques [Aura]

4. Conclusion

One option for binding self-certifying names to real-world identities is using a Web-of-Trust. This document aims at a concrete specification for providing such a binding, taking existing IETF specification into account. Future versions of this document will provide a more detailed specification.

5. References

5.1. Normative References

- [RFC2440] Callas, J., Donnerhacke, L., Finney, H., and R. Thayer, "OpenPGP Message Format", RFC 2440, November 1998.
- [RFC6920] Farrell, S., Kutscher, D., Dannewitz, C., Ohlman, B., Keranen, A., and P. Hallam-Baker, "Naming Things with Hashes", RFC 6920, April 2013.

5.2. Informative References

- [Aura] Aura, T. and M. Roe, "Strengthening Short Hash Values", <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.145.7681>, .
- [Aura2003] Aura, T., "Cryptographically Generated Addresses (CGA)", 6th International Conference on Information Security (ISC), 2003, .
- [Ghodsi2011] Ghodsi, A., Koponen, T., Rajahalme, J., Sarolahti, P., and S. Shenker, "Naming in Content-oriented Architectures", ACM SIGCOMM Workshop on Information-centric Networking, 2011, .
- [I-D.seedorf-icn-disaster] Arumaithurai, M., Seedorf, J., Tagami, A., Ramakrishnan, K., and N. Blefari-Melazzi, "Using ICN in disaster scenarios", draft-seedorf-icn-disaster-01 (work in progress), October 2013.
- [Nom2014] Seedorf, J., Kutscher, D., and F. Schneider, "Decentralised Binding of Self-Certifying Names to Real-World Identities for Assessment of Third-Party Messages in Fragmented Mobile Networks", 2nd Workshop on Name Oriented Mobility (NOM), 2014, .
- [Survey] Xylomenos, G., Ververidis, C., Siris, V., Fotiou, N., Tsilopoulos, C., Vasilakos, X., Katsaros, K., and G. Polyzos, "A Survey of Information-Centric Networking Research", IEEE Communications Surveys and Tutorials, Vol. 16, No. 2, pp 1024-1049, 2014, .

Appendix A. Acknowledgment

This document has been supported by the GreenICN project (GreenICN: Architecture and Applications of Green Information Centric Networking), a research project supported jointly by the European Commission under its 7th Framework Program (contract no. 608518) and the National Institute of Information and Communications Technology (NICT) in Japan (contract no. 167). The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the GreenICN project, the European Commission, or NICT.

Author's Address

Jan Seedorf
NEC
Kurfuerstenanlage 36
Heidelberg 69115
Germany

Phone: +49 6221 4342 221
Fax: +49 6221 4342 155
Email: seedorf@neclab.eu