ICNRG
Internet-Draft
Intended status: Informational
Expires: April 4, 2021

D. Trossen
Huawei
S. Robitzsch
InterDigital Inc.
M. Reed
M. Al-Naday
Essex University
J. Riihijarvi
RWTH Aachen

October 1, 2020

Internet Services over ICN in 5G LAN Environments
draft-trossen-icnrg-internet-icn-5glan-04

Abstract

In this draft, we provide architecture and operations for enabling Internet services over ICN over (5G-enabled) LAN environments. Operations include ICN API to upper layers, HTTP over ICN, Service Proxy Operations, ICN Flow Management, Name Resolution, Mobility Handling, and Dual Stack Device Support.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

As discussed in [I-D.irtf-icnrg-5gc-icn], Information-Centric Networks (ICN) could be more easily implemented in a Local Area Network (LAN) environment. In relation to 5G, this specifically would realize an ICN deployment without requiring integration of ICN capabilities into the 5G core network itself.

In the currently defined 5G core network, 5GLAN capabilities are being introduced that provide a LAN abstraction to 5G endpoints,

allowing for Ethernet packets to be sent across a 5G network, therefore extending the provisioning of LAN capabilities from fixed and Wifi-based networks to cellular ones.

Utilizing such ICN realization over 5GLAN, the objective of this draft is to propose an architecture to enable Internet services over such ICN-over-LAN environment with the reference architectural discussions in the 5G core network 3GPP specifications [TS23.501] [TS23.502] forming the basis of our discussions. This draft also complements work related to various ICN deployment opportunities explored in [RFC8763], where 5G technology is considered as one of the promising alternatives. In that, ICN is used as an underlay technology to provide routing capabilities to Internet services.

Through such replacement of IP routing with ICN routing, we capitalize on several ICN capabilities:

- Edge Computing: Multi-access Edge Computing (MEC) is located at the edge of the network and aids several latency sensitive applications such as augmented and virtual reality (AR/VR), as well as the ultra reliable and low latency class (URLLC) of applications such as autonomous vehicles. Enabling edge computing over an IP converged 5GC comes with the challenge of application level reconfiguration required to re-initialize a session whenever it is being served by a non-optimal service instance topologically. In contrast, named-based networking, as considered by ICN, naturally supports service-centric networking, which minimizes network related configuration for applications and allows fast resolution for named service instances. This opportunity is realized by interpreting Internet services as transactions over an ICN routed network with flexible routing to the nearest execution point for said transaction.

- Edge Storage and Caching: A principal design feature of ICN is the secured content (or named data) object, which allows location independent data replication at strategic storage points in the network, or data dissemination through ICN routers by means of opportunistic caching. These features benefit both real-time and non-real-time applications whenever there is spatial and temporal correlation among content accessed by these users, thereby advantageous to both high-bandwidth and low-latency applications such as conferencing, AR/VR, and non-real time applications such as Video-on-Demand (VOD) and IoT transactions. This opportunity is realized by the transaction-based model of realizing Internet service on top of an ICN routed network, where transaction results can be retrieved from a number of network locations.

- Opportunistic Multicast: The vast majority of current Internet traffic is due to unicast delivery of relatively immutable content such as video or software to very large client groups. This has resulted in large amount of redundancy in network traffic, as well as creating capacity bottlenecks both in the core network as well as the server infrastructure serving the content. Technologies such as content delivery networks (CDNs) help to spread out the network load, but are complex to manage, have inherent limits in terms of how rapidly they can react to changing network and server conditions, and cannot fundamentally reduce the network overhead arising from redundant unicast streams. Furthermore, CDNs traditionally only reach into Points-of-Presence (POP) within customer networks, therefore not reducing the load of transfer from said POP to the end customers in that edge network. In contrast, ICN enables opportunistic multicast delivery of content. We realize this opportunity by automatically delivering responses to quasi-concurrent requests in a single lightweight multicast transmission over the L2 customer network, extending the reach of CDNs down to the end user. Unlike traditional IP multicast, no setup time overhead is added and no per-flow state is required in the network.

In this document, we first outline possible use cases, capitalizing on the aforementioned ICN capabilities before discussing the proposed extensions to 5G to support a cellular-based LAN connectivity before outlining our proposal to support Internet services over an ICN-routed LAN connectivity in such 5G environments.

2. Terminology

Following are terminologies relevant to this draft:

5G-NextGen Core (5GC): Refers to the new 5G core network architecture being developed by 3GPP, we specifically refer to the architectural discussions in [TS23.501] [TS23.502].

5GLAN: Refers to the extensions to the new 5G core network architecture that provide LAN connectivity to 5G devices connected via, e.g., new 5G air interfaces.

User Plane Function (UPF): UPF is the generalized logical data plane function with context of the UE PDU session. UPFs can play many roles, such as, being a flow classifier, a PDU session anchoring point, or a branching point.

Packet Data Network (PDN or DN): This refers to service networks that belong to the operator or third party offered as a service to the UE.

Unified Data Management (UDM): Realizes unified data management for wireless, wireline and any other types of subscribers for M2M, IOT applications, etc. UDM reports subscriber related vital information e.g. virtual edge region, list of location visits, sessions active etc. UDM works as a subscriber anchor point so that means OSS/BSS systems will have centralized monitoring-of/ access-to of the system to get/set subscriber information.

Authentication Server Function (AUSF): Provides mechanism for unified authentication for subscribers related to wireless, wireline and any other types of subscribers such as M2M and IOT applications. The functions performed by AUSF are similar to HSS with additional functionalities to related to 5G.

Session Management Function (SMF): Performs session management functions for attached users equipment (UE) in the 5G Core. SMF can thus be formed by leveraging the Control and User Plane Separation (CUPS) feature with control plane session management.

Access Mobility Function (AMF): Perform access mobility management for attached user equipment (UE) to the 5G core network. The function includes, network access stratus (NAS) mobility functions such as authentication and authorization.

Application Function (AF): Helps with influencing the user plane routing state in 5GC considering service requirements.

Network Slicing: This conceptualizes the grouping for a set of logical or physical network functions with its own or shared control, data and service plane to meet specific service requirements.

3. Use Cases

3.1. 5G Control Plane Services

We exemplify the need for chaining service functions at the level of a service name through a use case stemming from the current 3GPP Rel-16 work on Service Based Architecture (SBA) [TS29.500] [SBA-ENHANCEMENT]. In this work, mobile network control planes are proposed to be realized by replacing the traditional network function interfaces with a fully service-based one. HTTP/2 was chosen as the application layer protocol for exchanging suitable service requests [TS29.500]. With this in mind, the exchange between, say the 3GPP (Rel-15) defined Session Management Function (SMF) and the Access and Mobility management Function (AMF) in a 5G control plane is being described as a set of web service like requests which are in turn embedded into HTTP requests. Hence, interactions in a 5G control

plane can be modelled based on service function chains where the relationship is between the specific service function endpoints that implement the necessary service endpoints in the SMF and AMF. The service functions are exposed through URIs with work ongoing to define the used naming conventions for such URIs.

This move from a network function model (in pre-Rel 15 systems of 3GPP) to a service-based model is motivated through the proliferation of data center operations for mobile network control plane services. In other words, typical IT-based methods to service provisioning, in particular that of virtualization of entire compute resources, are envisioned to being used in future operations of mobile networks. Hence, operators of such future mobile networks desire to virtualize service function endpoints and direct (control plane) traffic to the most appropriate current service instance in the most appropriate (local) data centre, such data centre envisioned as being interconnected through a software-defined wide area network (SD-WAN). 'Appropriate' here can be defined by topological or geographical proximity of the service initiator to the service function endpoint. Alternatively, network or service instance compute load can be used to direct a request to a more appropriate (in this case less loaded) instance to reduce possible latency of the overall request. Such data center centric operation is extended with the trend towards regionalization of load through a 'regional office' approach, where micro data centers provide virtualizable resources that can be used in the service execution, creating a larger degree of freedom when choosing the 'most appropriate' service endpoint for a particular incoming service request. This 5G control plane scenario capitalizes on the edge computing capabilities of ICN by allowing for fast redirections of HTTP-based transactions to the nearest control plane service realization within the distributed data centre of the 5G operator infrastructure.

3.2. HTTP-based Streaming

With the extensive use of "web technology", "distributed services" and availability of heterogeneous network, HTTP has effectively transitioned into the common transport or session layer for E2E and multi-hop communication across the web. Assume clients that are consuming the same content (such as a TV program) and that this content has for each block (typically segments worth 2 seconds of content) a set of outstanding requests from its clients. HTTP request and response used in media streaming services like HLS, use HTTP response for delivery of content. In such scenarios, where semi-synchronous access to the same resource occurs (such as watching prominent videos over Netflix or similar platforms or live TV over HTTP), traffic grows linearly with the number of viewers since the HTTP-based server will provide an HTTP response to each individual

viewer. To mitigate the load impact, operators often utilize IP multicast underneath HTTP (for live TV) to create fewer, multicast, streams; though this comes with the high flow setup and management cost. This poses a significant burden on operators in terms of costs and on users in terms of likely degradation of quality.

This problem is not limited to traditional TV broadcasting. Consider a virtual reality use case where several users are joining a VR session at the same time, e.g., centered around a joint event. Hence, due to the temporal correlation of the VR sessions, we can assume that multiple requests are sent for the same content at any point, particularly when viewing angles of VR clients are similar or the same. Due to availability of virtual functions and cloud technology, the actual end point from where content is delivered may change. For this type of scenarios, the opportunistic multicast capability of ICN may be utilized to reduce overall load in the network, as well as on the server providing the HTTP responses. The latter also allows constrained resources to serve a higher volume of demands and therefore incur a higher impact on traffic distribution in the network.

4. 5GLAN in 5G Next Generation Core Network Architecture

In this section, for brevity purposes, we restrict the discussions to the 5G extensions currently studied in 3GPP to facilitate a distributed, cellular-based LAN connectivity to end users, based on the 5G next generation core network architecture. For more information on the latter, we refer to [TS23.501] [TS23.502] as well as [I-D.irtf-icnrg-5gc-icn].

```
    +------+  +------+  +-----+   +-----+   +-----+   +-----+
    | NSSF |  | NEF  |  | NRF |   | PCF |   | UDM |   | AF  |
    +--o---+  +--o---+  +--o--+   +--o--+   +--o--+   +--o--+
  Nnssf|     Nnef|     Nnrf|     Npcf|     Nudm|      Naf|
  -----+-------+-+---------+--+------+-------+-+---------+---------
          Nausf|          Namf|          Nsmf|
            +--o--+        +--o--+        +--o--+
            | AUSF|        | AMF |        | SMF |
            +-----+        +-+-+-+        +--+--+
                            /  |             |
                 +---------+   |             |
            N1  /              |N2         N4|  +-N9/Nx-+
        +------+               |             |  |       |
       /                       |             |  |       V
    +-+--+                +----+----+  N3  +-+--+-------+--+  N6  +----+
    | UE +----------------+  (R)AN  +------+      UPF      +----->+ DN |
    +----+                +---------+      +---------------+      +----+
```

Figure 1: 5G Core Network with Vertical LAN (5GLAN) Extensions

Figure 1 shows the current 5G Core Network Architecture being discussed within the scope of the normative work addressing 5GLAN Type services in the 3GPP System Architecture Working Group 2 (3GPP SA2), referred formally as "5GS Enhanced support of Vertical and LAN Services" [SA2-5GLAN]. The goal of this work item is to provide distributed LAN-based connectivity between two or more terminals or User Equipment entities (UEs) connected to the 5G network. The SMF (session management function) provides a registration and discovery protocol that allows UEs wanting to communicate via a relevant 5GLAN group towards one or more UEs also members of this 5GLAN group, to determine the suitable forwarding information after each UE previously registered suitable identifier information with the SMF responsible to manage the paths across UEs in a 5GLAN group. UEs register and discover (obtain) suitable identifiers during the establishment of a Protocol Data Unit (PDU) Session or PDU Session Modification procedure. Suitable identifier information, according to [SA2-5GLAN], are Ethernet MAC addresses as well as IP addresses (the latter is usually assigned during the session setup through the SMF, i.e., the session management function).

The SMF that manages the path across UEs in a 5GLAN group, then establishes the suitable procedures to ensure the forwarding between the required UPFs (user plane functions) to ensure the LAN connectivity between the UEs (user equipments) provided in the original request to the SMF. When using the N9 interface to the UPF, this forwarding will rely on a tunnel-based approach between the UPFs along the path, while the Nx interface uses path-based forwarding

between UPFs, while LAN-based forwarding is utilized between the final UPF and the UE (utilizing the N3 interface towards the destination UE).

In the following, path-based forwarding is assumed, i.e., the usage of the Nx interface and the utilization of a path identifier for the end-to-end LAN communication. Here, the path between the source and destination UPFs is encoded through a bitfield, provided in the packet header. Each bit position in said bitfield represents a unique link in the network. Upon receiving an incoming packet, each UPF inspects said bitfield for the presence of any local link that is being served by one of its output ports. Such presence check is implemented via a simple binary AND and CMP operation. If no link is being found, the packet is dropped. Such bitfield-based path representation also allows for creating multicast relations in an ad hoc manner by combining two or more path identifiers through a binary OR operation. Note that due to the assignment of a bit position to a link, path identifiers are bidirectional and can therefore be used for request/response communication without incurring any need for path computation on the return path.

For sending a packet from one Layer 2 device (UE) connected to one UPF (via a RAN) to a device connected to another UPF, we provide the MAC address of the destination and perform a header re-write by providing the destination MAC address of the ingress UPF when sending from source device to ingress and placing the end destination MAC address in the payload. Upon arrival at the egress UPF, after having applied the path-based forwarding between ingress and egress UPF, the end destination address is restored while the end source MAC is placed in the payload with the egress L2 forwarder one being used as the L2 source MAC for the link-local transfer. At the end device (or proxy device), the end source MAC address is restored as the source MAC, providing an abstraction of a link-local L2 communication between the end source and destination devices.

```
      +---------+---------+----------+-----------+-----------+
      | Src MAC | Dst MAC |  pathID  |  NAME_ID  |  Payload  |
      +---------+---------+----------------------+-----------+
```

Figure 2: General Packet Structure

For this end-to-end transfer, the general packet structure of Figure 2 is used. The Name_ID field is being used for the ICN operations, while the payload contains the information related to the transaction-based flow management described in Section 5.8 and the

PATH_ID is the bitfield-based path identifier for the path-based forwarding.

4.1. Realization in SDN Transport Networks

An emerging technology for Layer 2 forwarding that suits the 5GLAN architecture in Figure 1 is that of Software-Defined networking (SDN) [SDN-DEFINITION], which allows for programmatically forwarding packets at Layer 2. Switch-based rules are being executed with such rules being populated by the SDN controller. Rules can act upon so-called matching fields, as defined by the OpenFlow protocol specification [OpenFlowSwitch]. Those fields include Ethernet MAC addresses, IPv4/6 source and destination addresses and other well-known Layer 3 and even 4 transport fields.

As shown in [Reed], efficient path-based forwarding can be realized in SDN networks by placing the aforementioned path identifiers into the IPv6 source/destination fields of a forwarded packet. Utilizing the IPv6 source/destination fields allows for natively supporting 256 links in a transport network. Larger topologies can be supported by extension schemes but are left out of this paper for brevity of the presentation. During network bootstrapping, each link at each switch is assigned a unique bitnumber in the bitfield. In order to forward based on such bitfield path information, the SDN controller is instructed to insert a suitable wildcard matching rule into the SDN switch. This wildcard at a given switch is defined by the bitnumber that has been assigned to a particular link at that switch during bootstrapping. Wildcard matching as a generalization of longest prefix matching is natively supported by SDN-based switches since the OpenFlow v1.3 specification, efficiently implemented through TCAM based operations. With that, SDN forwarding actions only depend on the switch-local number of output ports, while being able to transport any number of higher-layer flows over the same transport network without specific flow rules being necessary. This results in a constant forwarding table size while no controller-switch interaction is necessary for any flow setup; only changes in forwarding topology (resulting in a change of port to bit number assignment) will require suitable changes of forwarding rules in switches.

4.2. Realization in Other Transport Networks

Although we focus the methods in this draft on Layer 2 forwarding approaches and realization of Internet-over-ICN over a 5G LAN enabled network, path-based transport networks can also be established as an overlay over otherwise Layer 2 networks. For instance, the BIER (Bit Indexed Explicit Replication) [RFC8279] efforts within the Internet Engineer Task Force (IETF) establish such path-based forwarding

transport as an overlay over existing, e.g., MPLS networks. The path-based forwarding identification is similar to the aforementioned SDN realization although the bitfield represents ingress/egress information rather than links along the path.

Yet another transport network example is presented in [Khalili], utilizing flow aggregation over SDN networks. The flow aggregation again results in a path representation that is independent from the specific flows traversing the network.

The proposed traffic engineering extensions to BIER, presented in [I-D.ietf-bier-te-arch], directly align with the SDN-based realization presented in Section 4.1, by proposing the same bitposition per transport link assignment being used, resulting in BIER bitstrings in which a dedicated forwarding path is encoded as a unique bitpattern containing said bitpositions of the chosen forwarding links. The BIER-TE controller plays a similar role as the northbound SDN controller application utilized for the solution in Section 4.1.

5. Internet Services over ICN over 5GLAN

```
                   +-------------------------------+
                   |       Forwarding Network      |     .... Control
                   |  +--------------------------+ |
                   |  | .          NR          . | |     **** Data
                   |  +-.----------------------.-+ |
+--------------+   |    .                      .   |    +--------------+
|     App      |   |  +-.---------+  +---------.-+ |    |      App     |
+-----+----+---+   |  | . ******* |  | ******* . | |    +--------------+
|HTTP*|TCP*|IP.|   |  | . * UPF * |  | * UPF * . | |    |.IP|*TCP|*HTTP|
+----*+---*+--.+   |  +-.-*-----*-+  +-*-----*-.-+ |    +.--+*---+*----+
|ICN *    *   .|   |    . *     *      *     * .   |    |.   *    * ICN|
+----*----*---.+  +---+ . *     ********     * . +---+  +.---*----*----+
|L2  *    *   ....|RAN+.. *                  * ..+RAN|....   *    *    |
|    **********************                  **********************    |
+--------------+  +---+ . *                  * . +---+  +--------------+
                   |    . *                  * .   |
                   |  +-.-*-----+      +-----*-.+  |
                   +--| . *  RAN|------|RAN  * .|--+
                      +-.-*-----+      +-----*-.+
                        . *                  * .
                        . *******      ******* .
Legacy    Service       ....... *      * .......   Service
Device    Proxy               . *      * .         Proxy
+-----+ +-------------------+ . *      * . +-------------------+
|APP *| |    *********      | . *      * . |     **********    |
+----*+ +----*+      *      | . *      * . |     *      +*----+
|HTTP*| |HTTP*|*******      | . *      * . |     ********|*HTTP|
+----*+ +----*-+     *      | . *      * . |     *      +*----+
|TCP *| |TCP * |******      | . *      * . |     *******| * TCP|
+----*+ +----*--+   +*------+ . *      * . +-----*+   +---*----+  +-------+
|IP  *| |IP  *  |***|* ICN .| . *      * . |.ICN *|***|   *  IP|  | IP    |
+----*+ +----*--+---+*-----.+ . *      * . +.----*+---+---*----+  |Peering|
|L2  *| |    *   L2  *     .... *      * ....    *        *    |  |Network|
|    *********       ************      ***********        ************    |
+-----+ +-------------------+              +-------------------+  +-------+
```

Figure 3: Internet Services over ICN over 5GLAN

Figure 3 shows the protocol layering for realizing Internet protocols over an ICN over 5GLAN transport, assuming an end-to-end LAN connectivity provided by solutions such as 5GLAN.

Note that such LAN connectivity can also be found in environments such as localized LAN-based deployments in smart cities, enterprises and others, with the UPF representing, e.g., an SDN switch (utilizing the methods outlined in Section 4.1). Hence, the solutions described in this section also applies to those deployments.

Key to the approach is that Internet services are being interpreted as the main unit of transfer in the architecture shown in Figure 3. For this, any Internet service is treated as a Named Service Transaction (NST) which is in turn suitably routed over an ICN layer in one or more other devices. As a result of this name-based interpretation of any Internet service, the protocol stack in end devices flattens to four layers with Internet services and ICN, with ICN acting as a name-based routing layer for all IP protocols implemented atop, with Layer 1 and 2 realizing the end-to-end packet forwarding outlined in Section 4 (over a 5G environment) or a general LAN environment provided through WiFi or fixed Ethernet technologies.

The general ICN operations are presented in Section 5.1 before discussing the assumed (strawman) API to the ICN layer in Section 5.2, which is used in turn to define the mapping of HTTP transactions to operations at the ICN layer in Section 5.3 for the example of HTTP. As explained in that section, the ICN layer uses an interaction with the NR to register and discover HTTP-based services for determining the suitable end-to-end packet forwarding information.

Interfaces to legacy devices and peering networks are preserved through service proxy devices, which terminate a traditional Internet protocol stack communication and translate it into a resulting flat protocol transaction. Termination here can be based on well-known port numbers for specific Internet protocols, ultimately falling back to the IP datagram service being the minimal service being mapped. The operations of said service proxy devices is described in Section 5.5.

An important aspect of the architecture is the mapping of the end-to-end flow semantic established in many Internet services onto the flat protocol stack. Section 5.7 outlines the flow management that exists between the end devices.

The mapping of protocol identifiers onto ICN forwarding relations, i.e., the operations of the name resolver (NR), shown in Figure 3, is described in Section 5.8, followed by the procedures for handling mobility of service providers and consumers in Section 5.9. Finally, the support for dual-stack devices, not requiring a service proxy device yet being able to also connect to existing IP routed networks, is described in Section 5.10.

5.1. General Operations

The semantics of our name-based routing is that of a publish-subscribe system over a name. The intention to receive packets with a certain name is expressed through a subscription while sending

packets to a name is expressed through a publication. The matching of a sender to a receiver is realized through NR in Figure 3. The exact nature of the matching is defined through the semantics of the service and, therefore, through the nature of the name provided. For instance, HTTP and raw Internet services are matched to exactly one subscriber only, providing an anycast capability, while IP multicast services are matched against any subscriber (with the IP multicast address being the name).

Structured names are used with the root specific to the (Internet) service name, such as URL, and therefore deriving the matching semantics directly from the name.

The subscription to a name is realized through a registration protocol between end device and NR. Hence, any end device exposing a certain Internet service registers the suitable name with the NR, which in turn stores reachability information that is suitable for path calculation between the ingress and egress L2 forwarders between which the communication eventually will take place. In our current realization, we utilize shortest paths only although other link weights can be utilized for, e.g., delay-constrained and other policies.

In our realization, we use network domain unique host identifiers that are being assigned to end devices during the connectivity setup. Sending a packet of a given Internet service is realized through a discovery protocol, which returns a suitable pathID, i.e., the forwarding information between ingress and egress L2 forwarder, and the destination MAC address of the hosting end device. It is this pathID and MAC address that is being used in the general packet structure of Figure 2 to forward the packet to the destination.

To reduce latency in further communication, the forwarding information is locally cached at the end device, while the cached information is being maintained through path updates sent by the NR in case of hosting end devices having moved or de-registered, therefore avoiding stale forwarding information.

5.2. ICN API to Upper Layers

The operations of the ICN layer are exposed to upper layers in Figure 1 through the following API calls, being exemplary here for the further explanation of operations in the next sub-sections:

- conn = send(name, payload)

- send(conn, payload)

- conn = receive(name, &payload)

- receive(conn, &payload)

The first send() call is used for initiating a send operation to a name with a connection handle returned, while the second send() is used for return calls, using a connection parameters that is being received with the receive() call to an incoming connection or for subsequence outgoing calls after an initial request to a name has been made. A return send() is being received at the other (client) side through the second receive() call where the conn parameter is obtained by the corresponding send() call for the outgoing call. With these API functions, we provide means for providing name-based transactions with return responses association provided natively.

The conn parameter represents the bitfield used for path-based forwarding in the remote host case or the hash of the local MAC address in case of link-local connections.

5.3. HTTP over ICN

5.3.1. General Mapping Procedures

To realize the flat device nature, Internet service layers, such as the HTTP protocol stack or the TCP protocol stack, will need to be adapted to run atop this new API, implementing the semantics of the respective Internet protocol through suitable transactions at the name level. In the example of HTTP, the standard operations of DNS resolution for the server to be contacted and opening of a TCP (for HTTP/1.1 or HTTP/2) or UDP (for HTTP/3) socket are altogether replaced by a single send(FQDN, HTTP request) call; but the response will be sent by the server, which received the request through a receive(FQDN, &payload) call, using the returned conn parameter to send the response with the second send() API call. Note that the use of bidirectional pathIDs, no NR lookup is performed at the HTTP serving endpoint.

In the light of HTTP/3, the same mappings apply as already described above with the exception that the service proxy intercepts incoming UDP traffic only if it carries an HTTP/3 payload. If the payload is not HTTP/3, the mappings as described in Section Section 5.4 apply.

5.3.2. Realizing Ad-Hoc Multicast Responses for HTTP

The basis of a named service transaction allows to deliver the same HTTP responses to several requestees in efficient multicast (see [I-D.ietf-bier-multicast-http-response] for use cases in a BIER-based transport network environment).

This opportunity is realized by sending the same payload (i.e., an HTTP response to the same resource across a number of pending requests) through a combination of several conn parameters received in the incoming requests via the receive() function.

What is required in the HTTP stack implementation is a logic to decide that two or more outstanding requests are possible to be served by one response. For this, upon receiving an incoming request, the HTTP stack determines any outstanding request to the same resource. 'Same' here is defined as URI-specific combination of the request URI and URI-specific header fields, such as browsing agent or similar, called requestID in the following.

Once such determination is made that two requests are relating to the same resource, i.e., are having the same request ID, the HTTP stack maintains a temporary mapping of the request ID to the respective conn parameters delivered by the receive() call. Upon receiving the HTTP response from its application-level logic, the HTTP stack will generate the suitable send(conn, payload) call where the provided conn parameter is bitwise OR of all previously stored conn parameters received in the receive() call. The ICN layer will recognize the use of those ad-hoc created conn parameters and set the destination MAC address in the general packet structure of Figure 2 to the Ethernet broadcast MAC address as the destination address, leading to sending the response to all end devices at the egress L2 forwarders to which the response will be forwarded based on the combined conn parameter. Alternatively, one could request IEEE assignment for a specific Ethernet multicast address for this scheme instead of using the broadcast address. For the local end device to determine the relevance of the response received at the broadcast channel, the HTTP stack of the serving endpoint includes the aforementioned requestID into the payload of the packet (see Figure 2), while the originating endpoint maintains an internal table with the requestID of pending requests and its associated conn handle. If no matching requestID is found, the packet is not being delivered to the ICN layer of the incoming device. If a request is found, the ICN layer delivers the response via the receive() call, using the conn handle stored in the pending request table. Note that this filtering mechanism can easily be implemented in hardware upon standardizing the appropriate payload and header fields.

5.4. IP over ICN

For non-HTTP traffic, the service proxy uses the destination IP address of an incoming IP packet from an IP endpoint as the information identifier for the NR to find the suitable service proxy, which can reach the sought after IP endpoint. The usage of subnet masks and a longest prefix matching approach inside the NR is

foreseen for this matching process. In a 5GLAN scenario, service proxies on UEs serve a single IP endpoint (i.e. the UE itself) and therefore are represented by a /32 subnet mask for its IP address inside the NR in the list of subsribers. Service proxies that serve an entire local domain the subnet configured on the LAN interface of the service proxy is being communicated to the NR for matching purposes. The service proxy that serves the public internet is represented as a wildcard match for any IP address that is not served within the operator's network.

5.5. Service Proxy Operations

The service proxy in Figure 3 serves the integration of legacy devices, i.e., with regular IP protocol stack, and the interconnection to IP-based peering networks. It registers suitable identifiers with the NR to ensure the reception of (ICN) packets, while providing suitable protocol termination for the various supported protocols. For instance, for HTTP, the service proxy towards the peering network will register a wildcard name to the NR to receive any HTTP request not destined to a network-locally registered FQDN, operating as an HTTP proxy towards the peering network. Service proxies also register to the IP subnet they have been assigned on their local IP-based interface(s). The assignment is envisaged through an out-of-band address assignment scheme, i.e. DHCP [RFC2131] [RFC8415].

5.6. Support for Transport Layer Security

With a vast amount of networking intense HTTP traffic being encrypted, supporting HTTP over Transport Layer Security (TLS) [RFC8446] is of paramount importance to continue offering the benefits of routing the internet over 5GLAN utilising the ICN principles outlined in this document, in particular the ability to transparently change the service endpoint handling a HTTP request per HTTP transaction.

After the TCP session has been established by the client with the server, which is transparently intercepted by the service proxy and mapped to an inter service proxy ICN flow, the client initiates a TLS handshake aiming to establish a secure connection. When the service proxy receives the ClientHello message from the client it has two choices: act as a TLS endpoint or act as a TLS proxy.

If the service proxy has the TLS certificate/key for the FQDN provided in the TLS ClientHello message, it acts as a transparent TLS endpoint by intercepting the TLS sessions and implementing the TLS procedures described in [RFC8446]. If the client eventually sends a HTTP request over the TLS session the service proxy can decrypt it to

obtain the HTTP request in its entirety to perform the steps described above how to map HTTP over ICN (and vice versa). On the other side of the ICN flow the service proxy acts as a TLS client towards the actual IP service endpoint. One of the key benefits of service proxies acting as TLS endpoints is their ability to still offer opportunistic multicast, as TLS (similar to TCP) is fully intercepted at both edges of the ICN communication domain.

If the TLS certificate/key for the FQDN in the TLS ClientHello message is not available to the service proxy, TLS control messages between client and servers are left intact and routed to the most suitable service proxy that is subscribd to the FQDN. However, as the service proxy is not able to see HTTP transactions routing benefits of the decsribed steps such as opportunistic multicast and transparent interruption-free re-routing of HTTP transactiosn to a more suitable IP servce endpoint are not feasible. In such scenario, the TLS session must be restablished between the client and the server and service continuity cannot be offered. However, it is important to mention that the TLS proxy mode still improves over a plain TCP connection, as for the latter the IP address provided by a DNS is being used to determine the destined service proxy and not the FQDN of the HTTP request.

5.7. ICN Flow Management

For all protocol mappings described in this section, the payload taken from the (intercepted) layer is send as payload of the ICN packet, as illustrated in Figure 2. It can be observed that two resource management regimes are present, i.e. the application to service proxy communication (IP) and the inter service proxy (ICN) one. In the IP resource management regime, TCP friendliness governs the various transport protocols in use allowing a per flow fair usage of the available networking resources. However, the resource regime between service proxies does not have such requirement; thus, the corresponding ICN flow management and error control allows an independent improved resource regime that must not be TCP friendly.

For an independent inter service proxy resource management scheme that treats each *-over-ICN mapping equally, the notion of HTTP, TCP and IP transactions are being introduced with the goal to treat them equally and therefore ensuring resource fairness among them with the benefit of long lasting ICN flow relationships among service proxies.

```
                                          +--------------+    +-------+
                                       __ |Ad-Hoc Service|    |Flow   |
                                      /   |Proxy Flow    |    |Control|
                                     |    +--------------+    +-------+
    +--------+                       |           |                |
    | IP/UDP |                       |           |                |
    |Datagram|--------------|        |    +--------------+    +-------+
    +--------+              |        |    |Service Proxy |----|Flow   |
                            |        |    |    Flow      |    |Control|
    +--------+              |         \   +--------------+    +-------+
    |TCP/TLS |---------|    |          \         |
    |Stream  |         |    |           \        |
    +--------+         |    |            \       |
                    +-----------+         +--------------+    +-------+
                    |    IP     |---------|Service Proxy |----|Error  |
                    |Translation|         |Translation   |    |Control|
                    +-----------+         +--------------+    +-------+
    +-------------+    |    |
    |HTTP Request |----|    |
    +-------------+         |
                            |
    +-------------+         |
    |HTTP Response|---------|
    +-------------+
```

Figure 4: Mapping of IP Transactions onto Service Proxy Transactions and Flows

Figure 4 illustrates the ICN flow management for inter service proxy communication. As it shows, all traffic from IP endpoints are translated into a unified IP transaction and mapped to a service proxy transaction. The resulting service proxy flow constitutes a long-term relationship between two service proxies. For each service proxy flow, there exists a flow-specific flow control relationship, which maintains flow parameters such as send credits, timers for round-trip time (RTT) dependent mechanisms, error rate information and others. Such serivce proxy flow between two service proxies represent the edge-to-edge resource management regime desribed above. Each service proxy flow consists of one or more service proxy transactions, each of which comes with its own error control relationship that maintains information such as sequence numbers, outstanding packets, segmentation/reassembly information and others. For retransmissions, the error control relies on service proxy flow-specific flow control information, such as timers, RTT information etc. With such mapping from IP transactions onto a service proxy transaction that has its own error control mechanism, it has been achieved that the data originating from and destined to end-to-edge

resource management regimes can be reliably transferred over the service proxy-to-service proxy network. Combining all transactions under a single resource management relationship, represented by the combined flow control mechanism for a single flow between the service proxies, now establishes the inter service proxy resource management scheme. Any competition for resources among service proxies is now governed by said scheme between flows. Given that all transactions between specific service proxies are mapped into a single service proxy flow, fair resource sharing among all transactions can be ensured.

One crucial aspect of the HTTP-over-ICN mapping is the possibility of so-called ad-hoc multicast relations, i.e., the ability to send responses from one IP applications to more than one other IP application and therefore to more than one service proxy. In this case, the specific IP transaction (e.g. an HTTP response) is mapped onto a service proxy transaction that in turn is realized over more than one service proxy-to-service proxy flow. This flow is called ad-hoc service proxy flow. For those cases, the flow control for the ad-hoc service proxy flow will utilize parameters across the various involved service proxy flows, resulting in an one-to-many relationship between the specific flow control for the ad-hoc serivce proxy flow and the flow control(s) of the involved serivce proxy flow(s). Such combined parameters might be the maximum RTT timer or the lowest credit value, representing the least common dominator of the resources across all involved flows.

As mentioned before, a service proxy flow constitutes a long-term relationship between two service proxies. This relationship can established in multiple ways: an explicit setup might be used akin to that of TCP's three-way handshake. Alternatevely, an implicit transaction-based flow establishment might be used; in this case, the sending of an initial transaction between two service proxies results in the creation of an service proxy flow context between those two service proxies, which is being reused for any future transfer between those two service proxies, i.e., constituting a service proxy flow. Flow termination can be explicit based on a handshake protocol, where one service proxy, wishing to terminate the flow, signals this to the corresponding service proxy. Other embodiments foresee the destruction of the service proxy flow via timeout, e.g., removing any internal service proxy flow context information upon firing of an inactivity timeout. Combining this with an implicit transaction-based flow establishment would make the notion of a service proxy flow entirely that of an internal (service proxy flow context) data structure, which is created upon sending the first transaction to a service proxy which had previously not been contacted, while destroying said data structure upon the firing of the aforementioned inactivity timeout.

5.8. NR Operations

The NR in Figure 3 combines the operations of the SMF and the PMF in 5GLAN (see Figure 1), by allowing for registering IP protocol identifiers for discovery and subsequent path computation by resolving the destination(s) to a suitable pathID and destination MAC address for forwarding. This will require extensions to the operations of the SMF to allow for such higher layer identifiers to be registered (and discovered), in addition to the already supported Ethernet and IP addresses.

5.9. Mobility Handling

EDITOR NOTE: left for future draft updates.

5.10. Dual Stack Device Support

Figure 3 outlines a protocol stack for the user equipment that realizes Internet services on top of the proposed name-based routing layer as a single stack device. However, [I-D.irtf-icnrg-icn-lte-4g] outlines the possibility of supporting dual-stack devices for 4G LTE networks by allowing IP as well as ICN protocol stacks to be deployed with the operation of IP and ICN based applications. [I-D.irtf-icnrg-5gc-icn] outlines the same dual-stack device realization for a 5G ICN realization. For both environments, a convergence layer is described that selects the appropriate data path for each ICN or IP application, e.g., based on configuration per application (similar to selecting network interfaces such as WiFi over cellular).

As a possible data path selection, [I-D.irtf-icnrg-icn-lte-4g] and [I-D.irtf-icnrg-5gc-icn] envision the realization of Internet-over-ICN (Section 4.2 in [I-D.irtf-icnrg-icn-lte-4g]) in which the convergence layer would realize similar mapping functions as described in this draft. Hence, we foresee the utilization of such dual-stack devices connected to an Internet services over ICN over 5GLAN environment. When utilizing the service proxy, IP applications that are configured to use the IP data path only could still utilize the ICN-based forwarding in the network. In that case, functionality such as the opportunistic multicast in Section 5.3.2 would only reach up to the service proxy with unicast traffic continuing along the data path towards the user equipment.

6. Deployment Considerations

The work in [RFC8763] outlines a comprehensive set of considerations related to the deployment of ICN. We now relate the solutions proposed in this draft to the two main aspects covered in the

deployment considerations draft, namely the 'deployment configuration' (covered in Section 3 of [RFC8763]) that is being realized and the 'deployment migration paths' (covered in Section 4 of [RFC8763]) that are being provided.

The solutions proposed in this draft relate to those "deployment configuration" as follows:

- The realization of Internet service on top of an ICN routing capabilities, as proposed in Section 5, follows the "ICN-as-an-Underlay" categorization, interpreting the ICN routing as an underlay to the Internet services with the path-based forwarding being compatible with the 5GLAN forwarding capabilities currently discussed in 3GPP and therefore providing an underlay integration capability for the ICN forwarding used in the proposed solution.

- The deployment of 5GLAN based ICN capabilities can be realized following the "ICN-as-a-Slice" deployment configuration, i.e., the 5GLAN connectivity is provided to a "vertical 5G customer" which in turn provides the ICN capability over 5GLAN within said network (and compute) slice at the endpoints of the 5GLAN connectivity, as proposed in Section 3.

In relation of the 'deployment migration paths', the solutions in this draft relate as follows:

- The integration with the 5GLAN capability, as proposed in Section 5, facilitates "edge network migration" (interpreting the cellular sub-system here as an edge network albeit a possibly geographically large one.

- The single stack realization, as proposed in Figure 3, as well as the dual-stack deployment, as proposed in Section 5.10, facilitate "application and services migration" through not only supporting ICN applications but also Internet applications through the proposed Internet-over-ICN mapping in the terminal.

- The Internet over ICN over 5GLAN deployment, possibly combined with an ICN-as-a-Slice deployment, facilitates the "content delivery networks migration" through a deployment of Internet-over-ICN-based 5GLAN connected CDN elements in (virtualized) edge network nodes or POP locations in the customer (5G) network.

7. Conclusion

In this draft, we explored the feasibility of enabling Internet services directly over ICN network over (5G)LAN environments. We proposed the architecture and discussed corresponding operations of

mapping Internet services onto name-based transactions, with the specific example of HTTP-based transactions. We described the flow management, the realization of opportunistic multicast responses for HTTP as well as the realization of dual-stack user equipment. Future updates to the draft will provide more details to mobility handling. We also described the deployment scenario for supporting Internet services over ICN over 5GLAN.

8. IANA Considerations

This document requests no IANA actions.

9. Security Considerations

Editor Note: to be added in future drafts.

10. Acknowledgments

Work towards developing the solutions outlined in this draft have been funded under grants of the [H2020POINT] and [H2020FLAME] projects.

11. Informative References

[H2020FLAME]
H2020, "The FLAME Project", https://www.ict-flame.eu/ .

[H2020POINT]
H2020, "The POINT Project", https://www.point-h2020.eu/ .

[I-D.galis-anima-autonomic-slice-networking]
Galis, A., Makhijani, K., Yu, D., and B. Liu, "Autonomic Slice Networking", draft-galis-anima-autonomic-slice-networking-05 (work in progress), September 2018.

[I-D.ietf-bier-multicast-http-response]
Trossen, D., Rahman, A., Wang, C., and T. Eckert, "Applicability of BIER Multicast Overlay for Adaptive Streaming Services", draft-ietf-bier-multicast-http-response-04 (work in progress), July 2020.

[I-D.ietf-bier-te-arch]
Eckert, T., Cauchie, G., and M. Menth, "Tree Engineering for Bit Index Explicit Replication (BIER-TE)", draft-ietf-bier-te-arch-08 (work in progress), July 2020.

[I-D.irtf-icnrg-5gc-icn]
Ravindran, R., suthar, P., Trossen, D., Wang, C., and G. White, "Enabling ICN in 3GPP's 5G NextGen Core Architecture", draft-irtf-icnrg-5gc-icn-03 (work in progress), July 2020.

[I-D.irtf-icnrg-icn-lte-4g]
suthar, P., Stolic, M., Jangam, A., Trossen, D., and R. Ravindran, "Native Deployment of ICN in LTE, 4G Mobile Networks", draft-irtf-icnrg-icn-lte-4g-08 (work in progress), July 2020.

[I-D.muscariello-intarea-hicn]
Muscariello, L., Carofiglio, G., Auge, J., Papalini, M., and M. Sardara, "Hybrid Information-Centric Networking", draft-muscariello-intarea-hicn-04 (work in progress), May 2020.

[I-D.white-icnrg-ipoc]
White, G., Shannigrahi, S., and C. Fan, "Internet Protocol Tunneling over Content Centric Mobile Networks", draft-white-icnrg-ipoc-02 (work in progress), June 2019.

[Khalili] Khalili, R., Poe, W., Despotovic, Z., and A. Hecker, "Reducing State of SDN Switches in Mobile Core Networks by Flow Rule Aggregation", IEEE ICCCN 2016, Hawaii, USA, August 2016.

[OpenFlowSwitch]
Open Networking Foundation, available at https://www.opennetworking.org/wp-content/uploads/2014/10/openflow-switch-v1.5.1.pdf, "OpenFlow Switch Specification V1.5.1", 2018.

[Reed] Reed, M., AI-Naday, M., Thomos, N., Trossen, D., Petropoulos, G., and S. Spirou, "Stateless Multicast Switching in Software Defined Networks", IEEE ICC 2016, Kuala Lumpur, Maylaysia, 2016.

[RFC2131] Droms, R., "Dynamic Host Configuration Protocol", RFC 2131, DOI 10.17487/RFC2131, March 1997, <https://www.rfc-editor.org/info/rfc2131>.

[RFC7927] Kutscher, D., Ed., Eum, S., Pentikousis, K., Psaras, I., Corujo, D., Saucez, D., Schmidt, T., and M. Waehlisch, "Information-Centric Networking (ICN) Research Challenges", RFC 7927, DOI 10.17487/RFC7927, July 2016, <https://www.rfc-editor.org/info/rfc7927>.

[RFC8279] Wijnands, IJ., Ed., Rosen, E., Ed., Dolganow, A., Przygienda, T., and S. Aldrin, "Multicast Using Bit Index Explicit Replication (BIER)", RFC 8279, DOI 10.17487/RFC8279, November 2017, <https://www.rfc-editor.org/info/rfc8279>.

[RFC8415] Mrugalski, T., Siodelski, M., Volz, B., Yourtchenko, A., Richardson, M., Jiang, S., Lemon, T., and T. Winters, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 8415, DOI 10.17487/RFC8415, November 2018, <https://www.rfc-editor.org/info/rfc8415>.

[RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <https://www.rfc-editor.org/info/rfc8446>.

[RFC8763] Rahman, A., Trossen, D., Kutscher, D., and R. Ravindran, "Deployment Considerations for Information-Centric Networking (ICN)", RFC 8763, DOI 10.17487/RFC8763, April 2020, <https://www.rfc-editor.org/info/rfc8763>.

[SA2-5GLAN]
3gpp-5glan, "SP-181129, Work Item Description, Vertical_LAN(SA2), 5GS Enhanced Support of Vertical and LAN Services", 3GPP , http://www.3gpp.org/ftp/tsg_sa/TSG_SA/TSGS_82/Docs/SP-181120.zip.

[SBA-ENHANCEMENT]
3gpp-sba-enhancement, "S2-182904, New SID for Enhancements to the Service-Based 5G System Architecture.", 3GPP , February 2018 (http://www.3gpp.org/ftp/tsg_sa/WG2_Arch/TSGS2_126_Montreal/Docs/S2-182904.zip).

[SDN-DEFINITION]
Open Networking Foundation, available at https://www.opennetworking.org/sdn-definition/, "Software-Defined Networking (SDN) Definition", 2018.

[TS23.501]
3gpp-23.501, "Technical Specification Group Services and System Aspects; System Architecture for the 5G System; Stage 2 (Rel.15)", 3GPP , December 2018.

[TS23.502]
3gpp-23.502, "Technical Specification Group Services and System Aspects; Procedures for the 5G System; Stage 2 (Rel. 15)", 3GPP , January 2019.

[TS29.500]
3gpp-29.500, "Technical Realization of Service Based Architecture.", 3GPP , January 2018.

Authors' Addresses

Dirk Trossen
Huawei Technologies Duesseldorf GmbH
205 Hansallee
Duesseldorf 40549
Germany

Email: dirk.trossen@huawei.com
URI: http://huawei-dialog.de/

Sebastian Robitzsch
InterDigital Inc.
64 Great Eastern Street, 1st Floor
London EC2A 3QR
United Kingdom

Email: Sebastian.Robitzsch@InterDigital.com
URI: http://www.InterDigital.com/

Martin Reed
Essex University

Colchester
United Kingdom

Email: mjreed@essex.ac.uk
URI: https://www.essex.ac.uk/people/reedm58703/martin-reed

Mays Al-Naday
Essex University

Colchester
United Kingdom

Email: mfhaln@essex.ac.uk
URI: https://www.essex.ac.uk/people/alned81405/mays-al-naday

Janne Riihijarvi
RWTH Aachen

Aachen
Germany

Email: jariihij@googlemail.com
URI: https://www.inets.rwth-aachen.de/about-us/janne-riihijaervi/