

dmm
Internet-Draft
Intended status: Standards Track
Expires: November 16, 2018

S. Homma
K. Kawakami
NTT
A. Akhavain
Huawei Canada Research Centre
May 15, 2018

Co-existence of 3GPP 5GS and Identifier Locator Separation Architecture
[draft-homma-dmm-5gs-id-loc-coexistence-01](#)

Abstract

This document describes an approach to introduce Identifier Locator Separation architecture into 3GPP 5GS with low-impact on its specification, and shows the features and considerations of this approach.

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 <https://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 November 16, 2018.

Copyright Notice

Copyright (c) 2018 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 (<https://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 | 3 |
| 2. | Definition of Terms | 3 |
| 2.1. | Terms of ID-LOC Protocols | 4 |
| 2.2. | Terms of 5GS | 5 |
| 3. | Mechanism on Data Plane | 5 |
| 4. | Mechanisms on Control Plane | 10 |
| 4.1. | Model 1: Independent Control Planes | 11 |
| 4.2. | Model 2: Interworking Control Planes | 11 |
| 4.3. | Model 3: Integrated Control Planes | 12 |
| 5. | Features Analysis | 12 |
| 5.1. | Benefits | 12 |
| 5.2. | Issues | 12 |
| 6. | Security Considerations | 12 |
| 7. | IANA Considerations | 12 |
| 8. | Acknowledgement | 13 |
| 9. | Informative References | 13 |
| Appendix A. | Case Studies on Use of LISP | 14 |
| A.1. | UE-to-UE Communication | 15 |
| A.1.1. | Case A-1: UEs allocated different dUPF | 15 |
| A.1.2. | Case A-2: UEs allocated the same xTR | 17 |
| A.1.3. | Consideration of Case that UE Moves to under Another xTR | 18 |
| A.2. | UE-to-dDN Communication | 18 |
| A.2.1. | Case A-3: UE communicates with neighbor dDN | 18 |
| A.2.2. | Case A-4: UE communicates with non-neighbor dDN | 20 |
| A.3. | UE-to-cDN/Internet Communication | 22 |
| A.3.1. | Case A-5: UE communicates with cDN | 22 |
| Appendix B. | Case Studies on Use of ILA | 24 |
| B.1. | UE-to-UE Communications | 25 |
| B.1.1. | Case B-1: UEs allocated different dUPF | 25 |
| B.1.2. | Case B-2: UEs allocated the same ILA node | 26 |
| B.2. | UE-to-dDN Communication | 28 |
| B.2.1. | Case B-3: UE communicates with neighbor dDN | 28 |
| B.2.2. | Case B-4: UE communicates with non-neighbor dDN | 30 |
| B.3. | UE-to-cDN/Internet Communication | 32 |
| B.3.1. | Case B-5: Internet Communication | 33 |
| B.3.2. | Case B-6: Internet Communication | 34 |
| | Authors' Addresses | 36 |

1. Introduction

Identifier Locator Separation (ID-LOC) architecture aims to simplify management of network, devices, and sessions by employing two namespaces: Identifier for device's identity, and Locator for its location in the network.

ID-LOC architecture can be implemented by a dedicated protocol such as LISP, ILA, ILNP, etc. The control plane of such ID-LOC protocols can be combined with one of different encapsulation techniques such as GTP-U, SRv6, MPLS, etc. at data plane to provide a customized solution. Furthermore, regarding control plane of ID-LOC, it can optionally even take advantage of enhanced PUB/SUB capable distributed databases to circulate ID-LOC mapping relationships in the network.

ID-LOC protocols are also expected to be used for optimizing user-plane of mobile network

[[I-D.bogineni-dmm-optimized-mobile-user-plane](#)]. Different alternatives to introduce ID-LOC architecture into 3GPP 5GS(5th Generation System), are under consideration in related IETF WG such as DMM WG.

Introducing ID-LOC architecture into mobile network can involve modifications to 5GS architecture and specifications that might span over several 5GS releases.

Therefore, an approach that enables the introduction of ID-LOC architecture into 5GS without change of its specifications and supports migration path toward a native ID-LOC native network can be useful to operators. Here, ID-LOC native network refers to a network that employs the ID-LOC architecture as only mechanism for packet forwarding.

The document aims to describe one such approach and clarify different features, and benefits.

2. Definition of Terms

This section describes general terms of ID-LOC architecture. This document also refers definitions of 3GPP 5GS [[TS.23.501-3GPP](#)], and some of such terms which are used in this document are listed in this section.

The detailed specifications of LISP are described in [[RFC6830](#)], [[RFC6831](#)], [[RFC6832](#)], [[RFC6833](#)], [[RFC6836](#)], [[RFC7215](#)], [[RFC8061](#)], and [[RFC8111](#)]. Moreover, definitions and specifications of another

approach to introduce LISP into 3GPP 5GS is described in [[I-D.farinacci-lisp-mobile-network](#)].

The detailed specification of ILA are described in [[I-D.herbert-intarea-ila](#)].

The detailed specification of SRv6 are described in [[I-D.ietf-6man-segment-routing-header](#)].

2.1. Terms of ID-LOC Protocols

Device Identifier (ID): An ID is an identifier of host or end point such as UE or network function including VM instance, container, etc. In ID-LOC architecture, IP or MAC address, as unique value, is assigned to an end device. Values of the source and destination IP/MAC address fields of packets sent from end points are used as ID.

Locator (LOC): A LOC is an IPv4 or IPv6 address of a LOC-node. In the case of SRv6 it can be the LOC-node's local SID representing the segment for which the LOC-node is the segment termination node.

LOC-node; A LOC-node is a node that has a unique locator within a network domain, and has functionalities to obtain destination locator and to forward packets to the LOC-node which has the locator. This node has access to an ID-LOC mapping table and obtains destination locator by looking up destination ID (destination address of a data packet) from the mapping table. If ID of the received packet is not registered in its own mapping table, a LOC-node requests mapping information of the ID and the assigned locator to ID-to-LOC mapping database. Also a LOC-node forwards packet to a peered LOC node by encapsulation or conversion of the IP header field such as IP address field, and decapsulates or reconverts packets received from another LOC-node. Different implementations of ID-LOC architecture use different forwarding mechanisms. LISP, for example, uses IPv4/v6 header and LISP header for encapsulation, whereas ILA tightly couples itself with IPv6, and SRv6 uses IPv6 and SIDs (Segment Identifiers).

ID-to-LOC Mapping Database: An ID-to-LOC mapping database is a database which contains all known ID-to-LOC mappings within an ID-LOC domain. The mapping information is updated when an end point moves to under another LOC-node. This database is either owned by an individual system or each LOC-node. If an external system owns the database, each LOC-node has an interface to the system to send a request and receive mapping information.

ID-to-LOC Mapping Table: An ID-LOC mapping table is a table in a LOC-node that stores ID-to-LOC mapping information and it is used for obtaining destination LOC from ID of received packet. ID-to-LOC mapping table typically contains a small piece of database. The table is updated when the LOC-node receives a new ID-to-LOC mapping information from ID-to-LOC mapping database or some of 5GS system.

Mapping System: A mapping system is a function which stores ID-to-LOC mapping database. This system has interfaces to receive a request from a LOC-node and send an ID-to-LOC mapping to it. It may be composed of multiple components.

2.2. Terms of 5GS

User Plane Function (UPF): An UPF handles the user plane paths. An UPF is connected to SMF with N4 interface. More detailed information is described in [\[TS.23.501-3GPP\]](#). This document defines two types of UPF, Central UPF (cUPF) and Distributed UPF (dUPF). Their features are described in [Section 3](#)

Uplink Classifier (ULCL): An ULCL is an UPF functionality that aims at diverting Uplink traffic, based on filter rules provided by SMF, towards Data Network (DN).

Data Network (DN): A DN is a network where network functions and entities, including operator or 3rd party services, are deployed. This document defines two types of DN, Central DN (cDN) and Distributed DN (dDN). Their features are described in [Section 3](#).

Radio Access Network (RAN): A RAN is an access network where radio bearer sent by UEs traverse. A RAN encapsulate users' packets with GTP-U.

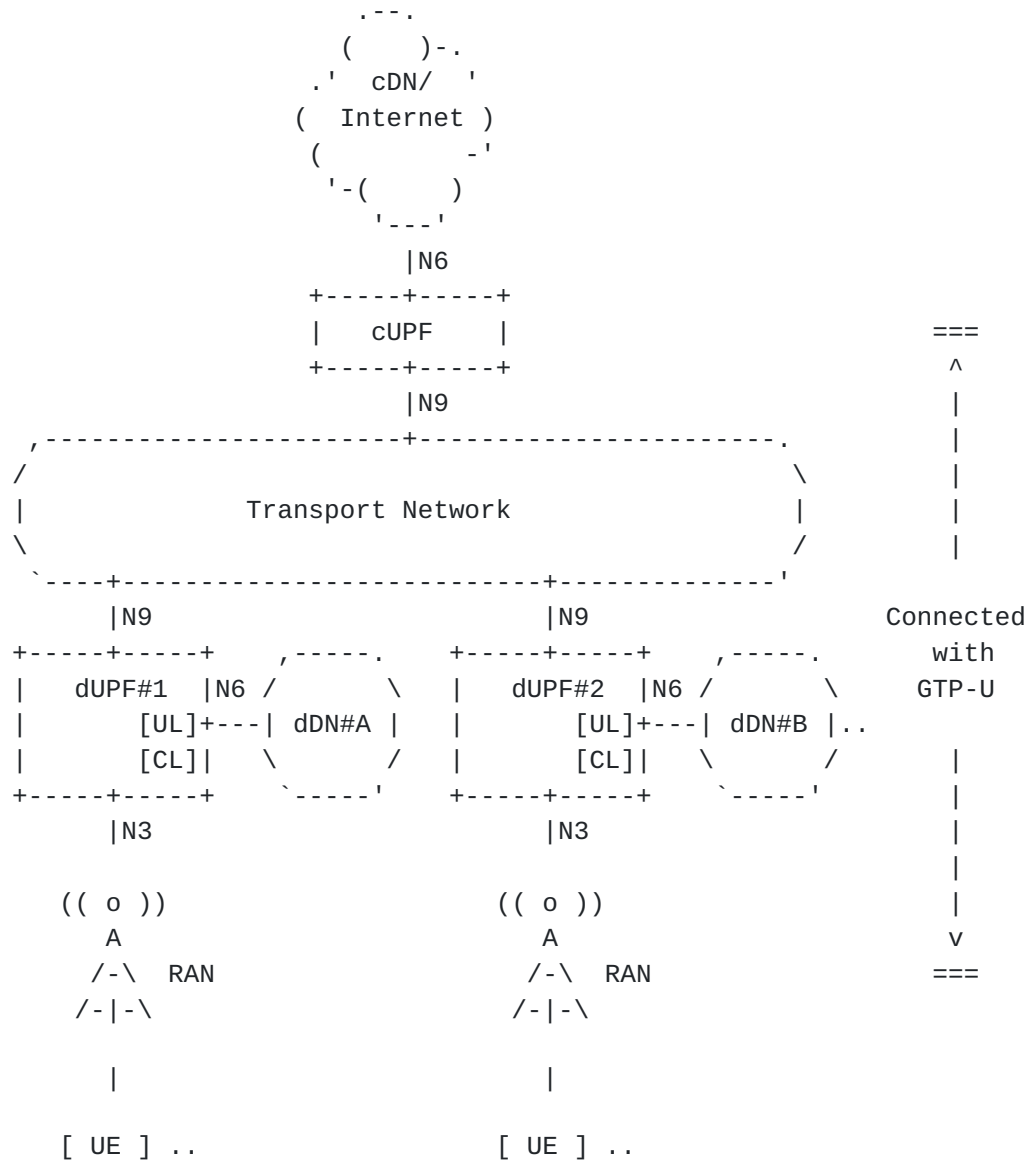
Session Management Function (SMF): An SMF is a function which provides control plane functionalities for handling user traffic.

Application Function (AF): An AF is a control plane functionality and connected to SMF with Naf interfaces.

3. Mechanism on Data Plane

This approach achieves traffic forwarding with optimized path and session continuity by using ID-LOC and ULCL for particular communication including UE-to-UE or MEC (Mobile Edge Computing) communication. ULCL is one of fundamental functions of 5GC Rel.15 and it provides functionalities of packet filtering and divert for uplink packets sent by UEs.

The overview of the assumed 5GC architecture of data plane where the proposal approach works is shown in Figure 1. The details of numbered interfaces in the figure are described in [TS.23.501-3GPP].



```
dUPF: Distributed UPF
cUPF: Central UPF
  dDN: Distributed DN
  cDN: Central DN
```

Figure 1: Assumed 5GC Network Architecture

This network has following features;

- o A Central UPF (cUPF) is deployed at a connecting point to Central DN (cDN). A cUPF becomes anchor point for UEs and it assigns IP addresses (IDs) for each UE. The traffic transmitted from UEs are basically sent to the cUPF.
- o Distributed UPFs (dUPFs) and Distributed DNS (dDNs) are deployed and geographically distributed at user edge side. A unique address space (it's not necessarily globally unique) is assigned to dDN. When a dUPF forwards an UE's uplink packet, and if the subnet of the destination address is the same as the one assigned to dDN at proximity, then dUPF, with the help of ULCL, may divert the packet to that dDN. Here, the ULCL identifies each encapsulated uplink packet to be diverted, by checking if the destination of the inner packet is one of IP addresses assigned the dDN. A dUPF removes GTP-U header from the packets, and sends them to dDN via N6. When dUPF receives packets from dDN, dUPF encapsulates them with GTP-U header, and merges them into downlink packets from cUPF. An overview of behaviors of dUPF and ULCL is shown in Figure 2.
- o Network topology between RAN and dUPF/cUPF adopts tree structure and the section between RAN and dUPF and the section between dUPF and cUPF are connected with GTP-U.

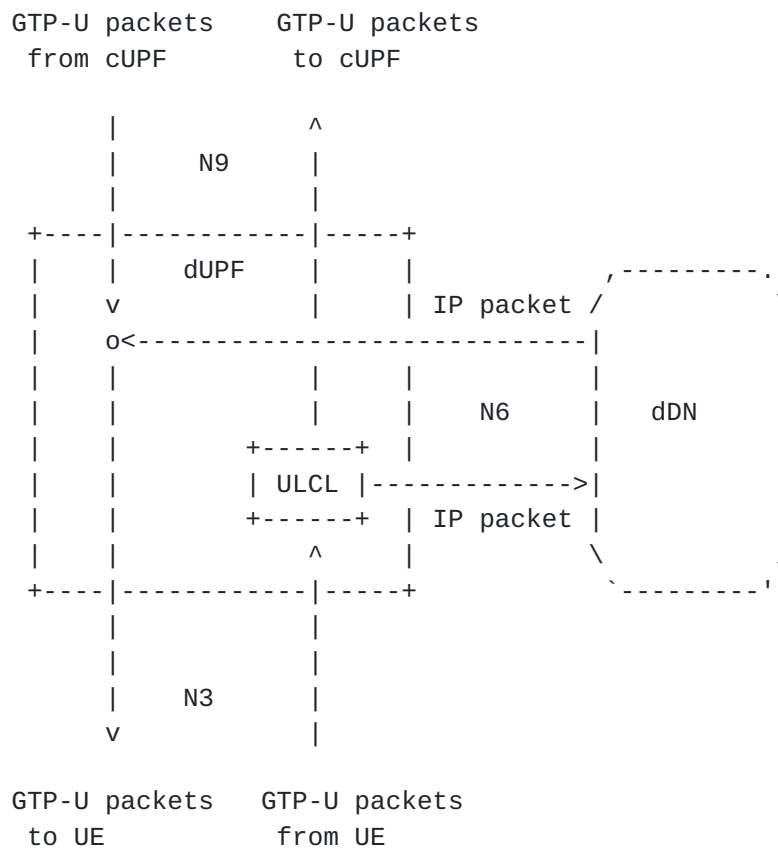


Figure 2: Behaviors of dUPF and ULCL

In the proposal approach, a LOC-node is installed between dUPF and dDN. LOC-nodes are connected with a IP mechanism such as IP tunnels or translation of destination IP field. As examples of such data plane protocols for providing connectivity between LOC-nodes, IPv4/v6 header with LISP header or SRv6

([\[I-D.ietf-6man-segment-routing-header\]](#)) can be used. In addition, each LOC-node has connectivity with one or more Mapping Systems. The overview is shown in Figure 3.

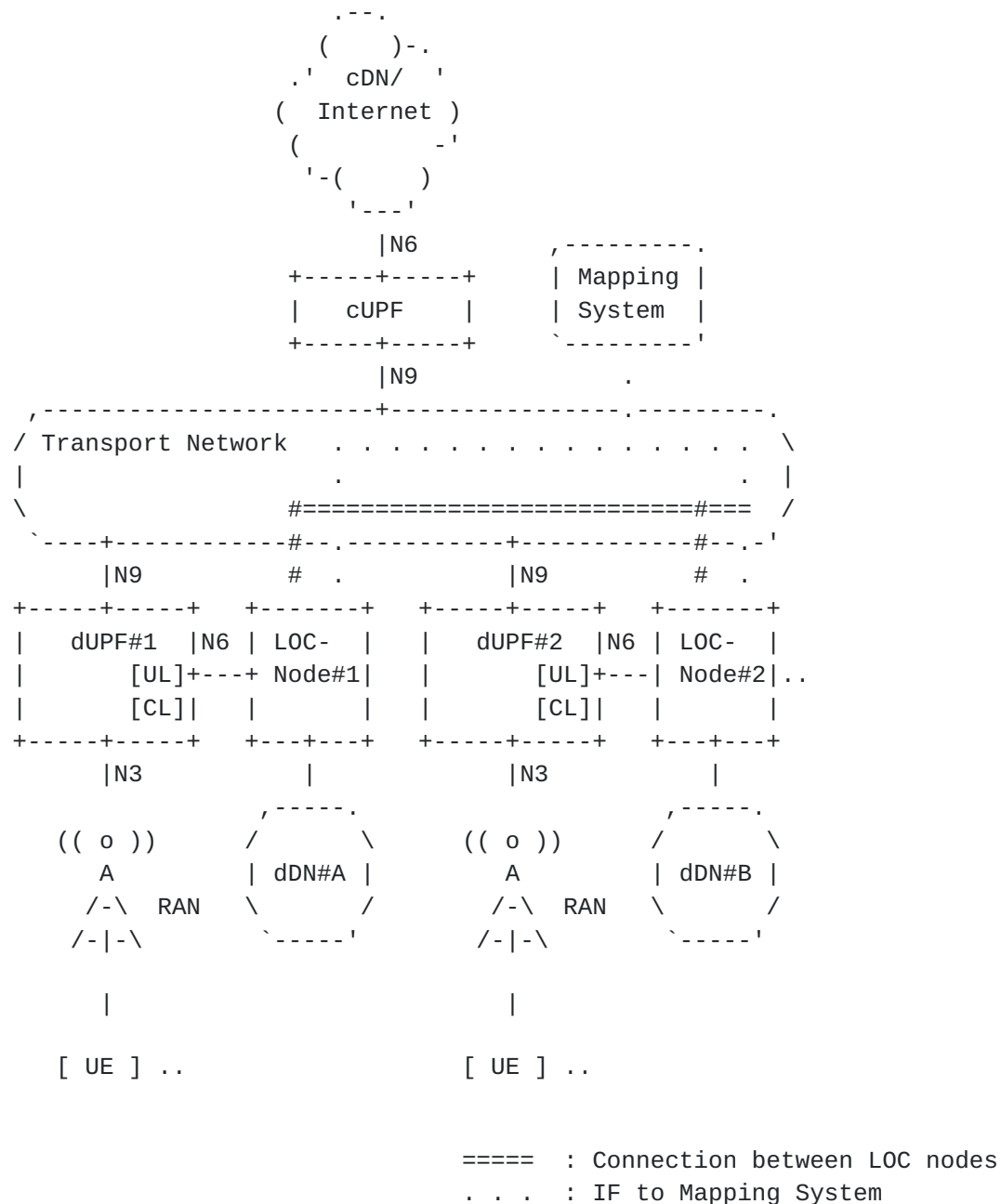


Figure 3: Proposal Network Architecture

Each dUPF has a filter table of ULCL. Each filter table is configured to match addresses assigned within own network domain (i.e., addresses for UEs assigned by cUPF) or assigned corresponding with address space of some of dDN. UPFs monitor each uplink GTP-U packet with its ULCL and divert it to the connected LOC-node with decapsulation of GTP-U if the destination address of the inner packet (payload) matches the filtering table. When LOC-node receives a

packet from the dUPF, it obtains LOC which the destination of the packet (ID) belongs to by looking up its own ID-to-LOC mapping table or querying it from the Mapping System according ID-LOC mechanism. Then it sends the packet to peered LOC-node indicated by the LOC. The peered LOC-node converts the received packet to appropriate form and forwards them the destination by following its own forwarding table.

From such processes, forwarding paths of user traffic diverted by ULCL from 5GC to LOC-node are optimized.

A cUPF is connected with dUPFs via N9 interface and packets are forwarded with GTP-U encapsulation between cUPF and dUPF.

Some case studies of ID-LOC protocols are described in [Appendix A](#) and [Appendix B](#).

4. Mechanisms on Control Plane

For ID-LOC mechanism in mobile networks, a control plane mechanism to manage location information of UEs is required. There are mainly three models to realize control plane mechanism for ID-LOC as follows:

Model 1: Independent Control Planes

Model 2: Interworking Control Planes

Model 3: Integrated Control Planes

Some of models may require to use 5GS interfaces or add some functionalities to functions of 5GC. 5GS architecture and the service-based interfaces are shown in Figure 4. The details of functions and interfaces are described in [[TS.23.501-3GPP](#)].

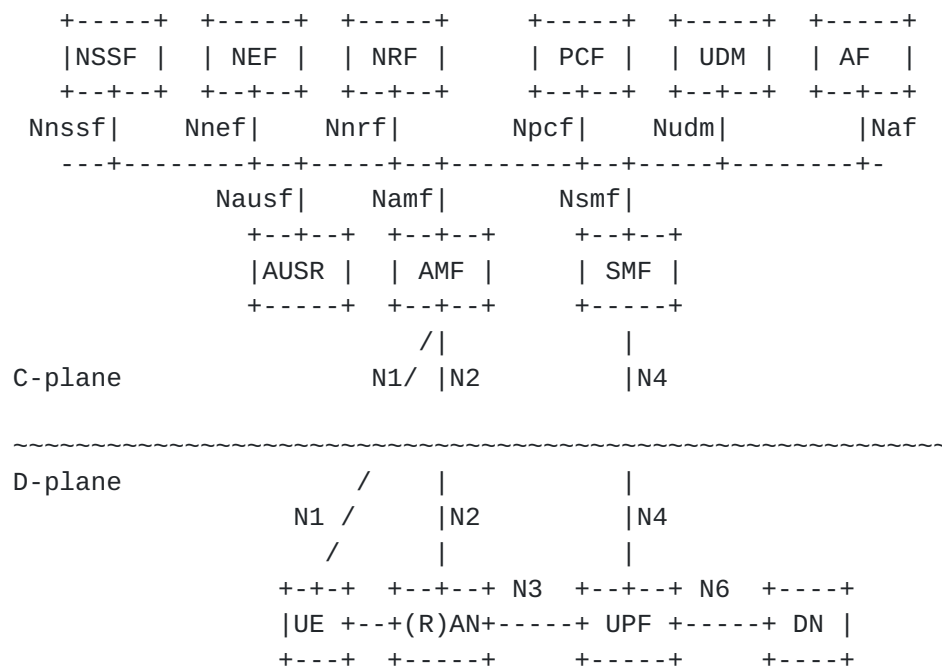


Figure 4: 5GS Architecture and Service-based Interfaces

4.1. Model 1: Independent Control Planes

In this model, control plane of 5GC and ID-to-LOC mapping mechanism are completely separated. Information of an UE and a LOC-node which the UE is attached is sent to a mapping system and registered in the mapping database only when the LOC-node receives a packet from the UE and the UE is not registered yet.

This model does not cause any impacts on 5GC architecture. However, in this model, an UE cannot be accessed from other UEs within the same network domain until a packet from the UE is diverted to the LOC-node by the UPF which the UE is located and the EID and RLOC are registered to the Mapping System.

4.2. Model 2: Interworking Control Planes

In this model, a mapping system interworks with an SMF which manages sessions of each UE. A scheme to inform, that an UE moves and is relocated to another UPF, from SMF to AF via Naf interface is defined in 5GS ([[TS.23.502-3GPP](#)])*. A Mapping System is installed as an AF and obtains mobility information of UEs with the above scheme.

* The stage 3 of discussion of 5GS has not been fixed yet and the specification may be changed.

This model would not cause any impacts on 5GS architecture, and a mapping system can always keep the current mobility information of each UE.

4.3. Model 3: Integrated Control Planes

In this model, SMF functionalities are integrated into a mapping system. In other words, the mapping system becomes a part of 5GS. In 5GS architecture, an SMF has a role of session management of UEs, and it updates its own mapping database depending on movement of an UE.

This approach enables to always keep mapping databases the latest status, however, it obviously requires extension or replacement of SMF actually deployed in 5GS network.

5. Features Analysis

5.1. Benefits

- o This approach provides a mechanism for introducing ID-LOC architecture into 5GS with no or nominal impact, and achieves optimized forwarding with session continuity in the assumed use cases such as UE-to-UE or UE-to-dDN communications.
- o Regarding communication to the cDN, this approach can keep scalability because it does not change the current mechanism of 5GS. (ID-LOC-native network or full-overlay approaches need to deploy LOC-node at the cUPF, and thus the ID-to-LOC mapping table may not scale up enough in that cases. Here, a full-overlay approach means making an ID-LOC system run over the whole 5GC network.)

5.2. Issues

- o dUPF and LOC-node are separated, and thus an extra hop may occur against the optimized forwarding. However, it can be resolved by implementing dUPF and LOC-node within a same box or application.

6. Security Considerations

TBD

7. IANA Considerations

This memo includes no request to IANA.

8. Acknowledgement

The authors would like to thank Ryosuke Kurebayashi, Koji Tsubouchi, Toru Okugawa, and Dino Farinacci for their kind reviews and technical feedback.

9. Informative References

- [I-D.bogineni-dmm-optimized-mobile-user-plane]
Bogineni, K., Akhavain, A., Herbert, T., Farinacci, D., and A. Rodriguez-Natal, "Optimized Mobile User Plane Solutions for 5G", [draft-bogineni-dmm-optimized-mobile-user-plane-00](#) (work in progress), March 2018.
- [I-D.farinacci-lisp-mobile-network]
Farinacci, D., Pillay-Esnault, P., and U. Chunduri, "LISP for the Mobile Network", [draft-farinacci-lisp-mobile-network-02](#) (work in progress), September 2017.
- [I-D.herbert-intarea-ila]
Herbert, T. and P. Lapukhov, "Identifier-locator addressing for IPv6", [draft-herbert-intarea-ila-00](#) (work in progress), October 2017.
- [I-D.ietf-6man-segment-routing-header]
Previdi, S., Filsfils, C., Raza, K., Dukes, D., Leddy, J., Field, B., daniel.voyer@bell.ca, d., daniel.bernier@bell.ca, d., Matsushima, S., Leung, I., Linkova, J., Aries, E., Kosugi, T., Vyncke, E., Lebrun, D., Steinberg, D., and R. Raszuk, "IPv6 Segment Routing Header (SRH)", [draft-ietf-6man-segment-routing-header-08](#) (work in progress), January 2018.
- [I-D.ietf-lisp-eid-mobility]
Portoles-Comeras, M., Ashtaputre, V., Moreno, V., Maino, F., and D. Farinacci, "LISP L2/L3 EID Mobility Using a Unified Control Plane", [draft-ietf-lisp-eid-mobility-00](#) (work in progress), May 2017.
- [RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The Locator/ID Separation Protocol (LISP)", [RFC 6830](#), DOI 10.17487/RFC6830, January 2013, <<https://www.rfc-editor.org/info/rfc6830>>.
- [RFC6831] Farinacci, D., Meyer, D., Zwiebel, J., and S. Venaas, "The Locator/ID Separation Protocol (LISP) for Multicast Environments", [RFC 6831](#), DOI 10.17487/RFC6831, January 2013, <<https://www.rfc-editor.org/info/rfc6831>>.

- [RFC6832] Lewis, D., Meyer, D., Farinacci, D., and V. Fuller, "Interworking between Locator/ID Separation Protocol (LISP) and Non-LISP Sites", [RFC 6832](#), DOI 10.17487/RFC6832, January 2013, <<https://www.rfc-editor.org/info/rfc6832>>.
- [RFC6833] Fuller, V. and D. Farinacci, "Locator/ID Separation Protocol (LISP) Map-Server Interface", [RFC 6833](#), DOI 10.17487/RFC6833, January 2013, <<https://www.rfc-editor.org/info/rfc6833>>.
- [RFC6836] Fuller, V., Farinacci, D., Meyer, D., and D. Lewis, "Locator/ID Separation Protocol Alternative Logical Topology (LISP+ALT)", [RFC 6836](#), DOI 10.17487/RFC6836, January 2013, <<https://www.rfc-editor.org/info/rfc6836>>.
- [RFC7215] Jakab, L., Cabellos-Aparicio, A., Coras, F., Domingo-Pascual, J., and D. Lewis, "Locator/Identifier Separation Protocol (LISP) Network Element Deployment Considerations", [RFC 7215](#), DOI 10.17487/RFC7215, April 2014, <<https://www.rfc-editor.org/info/rfc7215>>.
- [RFC8061] Farinacci, D. and B. Weis, "Locator/ID Separation Protocol (LISP) Data-Plane Confidentiality", [RFC 8061](#), DOI 10.17487/RFC8061, February 2017, <<https://www.rfc-editor.org/info/rfc8061>>.
- [RFC8111] Fuller, V., Lewis, D., Ermagan, V., Jain, A., and A. Smirnov, "Locator/ID Separation Protocol Delegated Database Tree (LISP-DDT)", [RFC 8111](#), DOI 10.17487/RFC8111, May 2017, <<https://www.rfc-editor.org/info/rfc8111>>.
- [TS.23.501-3GPP]
3rd Generation Partnership Project (3GPP), "3GPP TS 23.501", December 2017, <http://www.3gpp.org/ftp//Specs/archive/23_series/23.501>.
- [TS.23.502-3GPP]
3rd Generation Partnership Project (3GPP), "3GPP TS 23.502", December 2017, <http://www.3gpp.org/ftp//Specs/archive/23_series/23.502>.

Appendix A. Case Studies on Use of LISP

This Appendix describes detailed processes of the proposal approach with LISP mechanism in the following types of communications.

1. UE-to-UE Communication

2. UE-to-dDN Communication

3. UE-to-cDN/Internet Communication

In the following description of case studies, ID and Locator are called EID (End-point Identifier) and RLOC (Routing Locator) in LISP terms. Mapping Server has the master of EID-to-RLOC mapping database, and each xTR (Ingress/Egress Tunnel Router) has EID-to-RLOC mapping cache. An xTR obtains the destination RLOC from its own cache by looking up the destination EID of received packet. They obtain mappings from the mapping system if an EID looked up is not registered in the cache. Packets are passed between xTRs with some tunnel protocols.

A.1. UE-to-UE Communication

In the current architecture, a cUPF becomes an anchor point for UEs, and all packets between UEs even those which are located to the same dUPF are transferred through the anchor point. This may cause communication delay and inefficient resource usage. In the proposed procedure, packets can be transferred without going through an anchor point, and low latency and efficient resource usage can be achieved.

The UE-to-UE communications include communications between UEs located to different dUPFs (Case 1), and communication between UEs located to the same dUPF (Case 2). In this section, the detailed procedures of the cases are described.

Moreover, in a mobile network, an UE may move during communications. This section describes problems and considerations about UE's handover in such case.

A.1.1. Case A-1: UEs allocated different dUPF

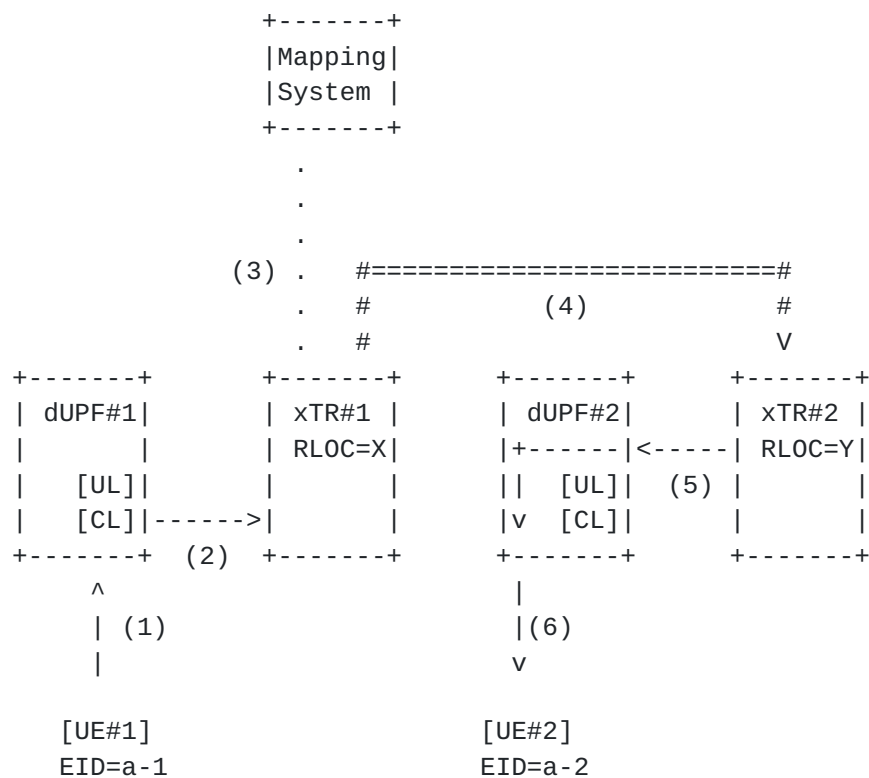


Figure 5: Procedure in Case A-1

- (0) Within this network, addresses are assigned to UEs from a address space [A]. These addresses are described as a-n (n=1,2,..). EID=a-1 and a-2 are assigned to UE#1 and UE#2.
- (1) UE#1 sends packets to UE#2 with setting EID=a-2 as the destination IP address.
- (2) dUPF#1 monitors inner packet of received GTP-U packet and divert it to xTR#1 with decapsulation if the destination address is one of address space [A].
- (3) xTR#1 updates own EID-to-RLOC mapping chace by interaction with Mapping System (if needed).
- (4) xTR#1 obtains the RLOC(=Y) of EID=a-2 from the EID-to-RLOC mapping cache, and sends the packets to the xTR#2 with a tunnel with RLOC=Y as the destination address.
- (5) xTR#2 decapsulate the packets, and sends them to dUPF#2.
- (6) dUPF#2 encapsulate packets with GTP-U header, and sends them to UE#2.

A.1.2. Case A-2: UEs allocated the same xTR

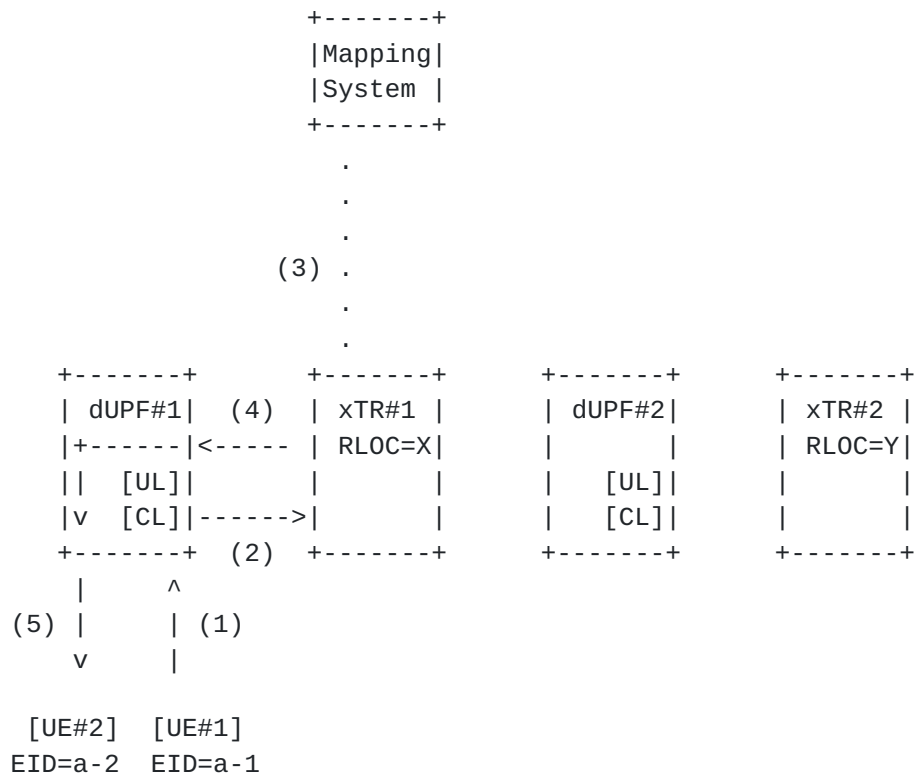


Figure 6: Procedure in Case A-2

- (0) Within this network, addresses are assigned to UEs from a address space [A] These addresses are described as a-n (n=1,2,...). EID=a-1 and a-2 are assigned to UE#1 and UE#2.
- (1) UE#1 sends packets to UE#2 with setting EID=a-2 as the destination IP address.
- (2) dUPF#1 monitors inner packets of received GTP-U traffic and divert it to xTR#1 with decapsulation if the destination address is one of address space [A].
- (3) xTR#1 updates own EID-to-RLOC mapping cache by interaction with Mapping System (if needed).
- (4) xTR#1 obtains the RLOC(=X) from the EID-to-RLOC mapping cache. Since RLOC=X indicates itself, xTR#1 sends the packets back to dUPF#1.
- (5) dUPF#2 encapsulate packets with GTP-U, and sends them to UE#2.

A.1.3. Consideration of Case that UE Moves to under Another xTR

When an UE moves to a serving area of another dUPF during communication with another UE, EID-to-RLOC mapping database of a Mapping System and the tables of the xTR and the peered xTR must be updated. The xTRs can't send packets to the appropriate xTR during the updating, and thus packet drop or stalling may occur.

To mitigate this problem, further consideration is needed. For example, a mechanism that immediately advertise the update of location of UEs to Mapping System and the appropriate xTRs depending on movement of each UE might be required. Also, some documents (e.g., [[I-D.ietf-lisp-eid-mobility](#)]) discuss this problem.

A.2. UE-to-dDN Communication

The UE-to-dDN communications basically correspond the communication between an UE and neighbor dDN (Case3). On the other hand, if an UE moved under another dUPF during usage of a statefull application, or the application is not uniformly deployed in every dDN, the UE needs to continue to communicate with the previous dDN (Case4).

In such cases, in the current architecture, all packets are needed to go through the anchor point or dynamic GTP tunnel reconfiguration between dUPF is required. The former solution causes additional communication delay and inefficient resource usage. The latter solution increase the cost of 5GS control plane to dynamically update the GTP tunnel with multiple UPFs and their ULCL filter tables along with the movement of the UE. The propal approach achieves appropriate packet transfer in such cases.

In this section, the detailed procedures of communications between an UE and neighbor dDN and communications between an UE and non-neighbor dDN

A.2.1. Case A-3: UE communicates with neighbor dDN

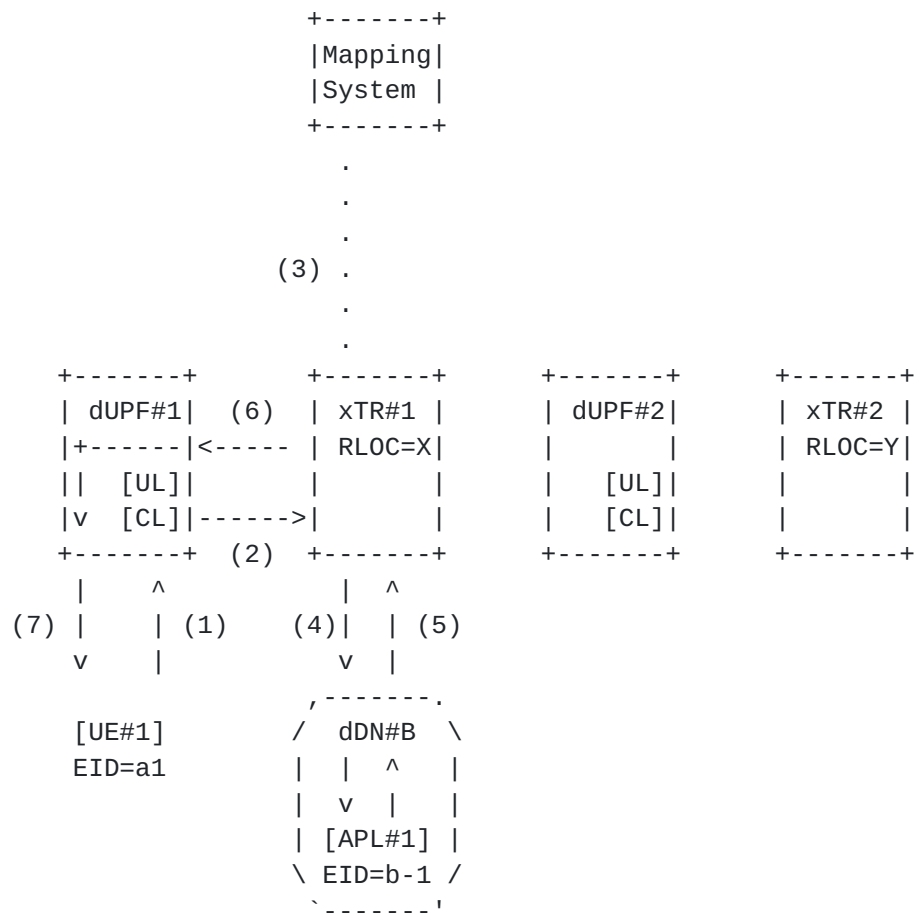


Figure 7: Procedure in Case A-3

- (0) Within this network, UEs are assigned their addresses from an address space [A]. These addresses are described as a-n (n=1,2,...). Also, applications in dDN#B are assigned their addresses from a address space [B]. These addresses are described as b-n (n=1,2,...). EID=a-1 and b-1 assigned to UE#1 and APL#1 which is located in dDN#B.

[Uplink Processes]

- (1) UE#1 sends packets to dDN#B with setting EID=b-1 as the destination IP address.
- (2) dUPF#1 monitors inner of received GTP-U packets and divert it to xTR#1 with decapsulation if the destination IP address is one of address space [B].

- (3) xTR#1 updates own EID-to-RLOC mapping cache by interaction with Mapping System (if needed). Or xTR#1 may update its own cache by a Map-Notify message when an APL is deployed or deleted in dDB#B.
- (4) xTR#1 obtains RLOC(=X) of EID=b-1 from the EID-to-RLOC mapping cache. Since RLOC=X indicates itself and EID=b-1 is within [B], xTR#1 sends the packets to the dDN#B.

[Downlink Processes]

- (5) APL#1 in dDN#B sends packets to UE#1 with setting EID=a-1 as the destination IP address.
- (6) xTR#1 obtains RLOC of EID=a-1 (i.e., RLOC=X) from the EID-to-RLOC mapping cache. Since RLOC=X indicates xTR#1 itself, xTR#1 sends packets to dUPF#1.
- (7) dUPF#2 encapsulates packets with GTP-U, and sends them to UE#1.

A.2.2. Case A-4: UE communicates with non-neighbor dDN

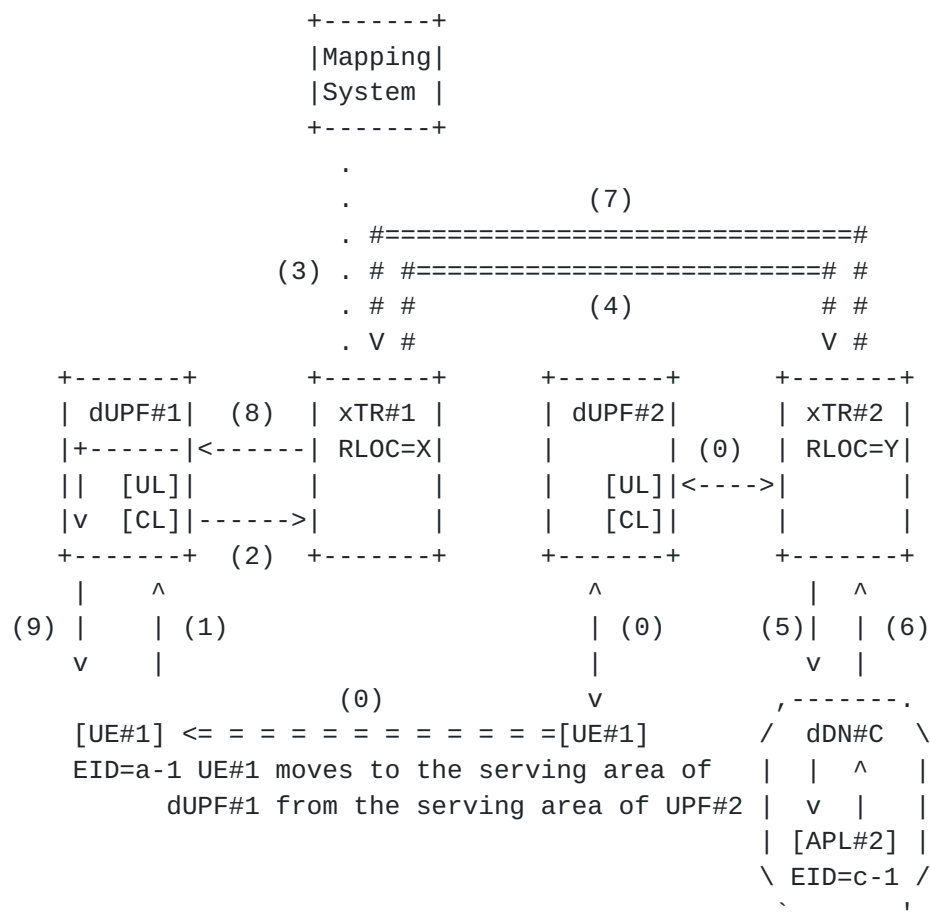


Figure 8: Procedure in Case A-4

- (0) Within this network, UEs are assigned their addresses from an address space [A]. These addresses are described as a-n (n=1,2,...). And applications in dDN#C are assigned their addresses from an address space [C]. These addresses are described as c-n (n=1,2,...). EID=a-1 and c-1 assigned to UE#1 and APL#2 which is located in dDN#C. UE#1 has moved to the serving area of dUPF#1 from the serving area of UPF#2 while communicating to APL#2.

[Uplink Processes]

- (1) UE#1 sends packets to APL#2 with setting EID=c-1 as the destination IP address.
- (2) dUPF#1 monitors each inner packet of received GTP-U traffic and divert it to xTR#1 with decapsulation if the destination address is one of address space [C].

- (3) xTR#1 updates own EID-to-RLOC mapping cache by interaction with Mapping System (if needed).
- (4) xTR#1 obtains RLOC(=Y) of EID=c-1 from the EID-to-RLOC mapping cache, and sends the packet to the xTR#2 with a tunnel with RLOC=Y as the destination address.
- (5) xTR#2 decapsulates the packets received from xTR#1, and sends them to dDN#C depending on its forwarding table.

[Downlink Processes]

- (6) APL#2 sends packets to UE#1 with setting EID=a-1 as the destination IP address.
- (7) xTR#2 obtains RLOC(=X) of EID=a-1 from the EID-to-RLOC mapping cache, and sends the packets to the xTR#1 with a tunnel with RLOC=X as the destination address.
- (8) xTR#1 decapsulates the packets received from xTR#2m and sends them to the dUPF#1 depending on its forwarding table.
- (9) dUPF#1 encapsulates the packets with GTP-U and sends packets to UE#1.

A.3. UE-to-cDN/Internet Communication

UE-to-cDN/Internet communication is achieved by GTP-U mechanism originally equipped in 3GPP 5GS architecture. In this section, we describe processes of UE-to-cDN communication in the proposal architecture as an example.

A.3.1. Case A-5: UE communicates with cDN

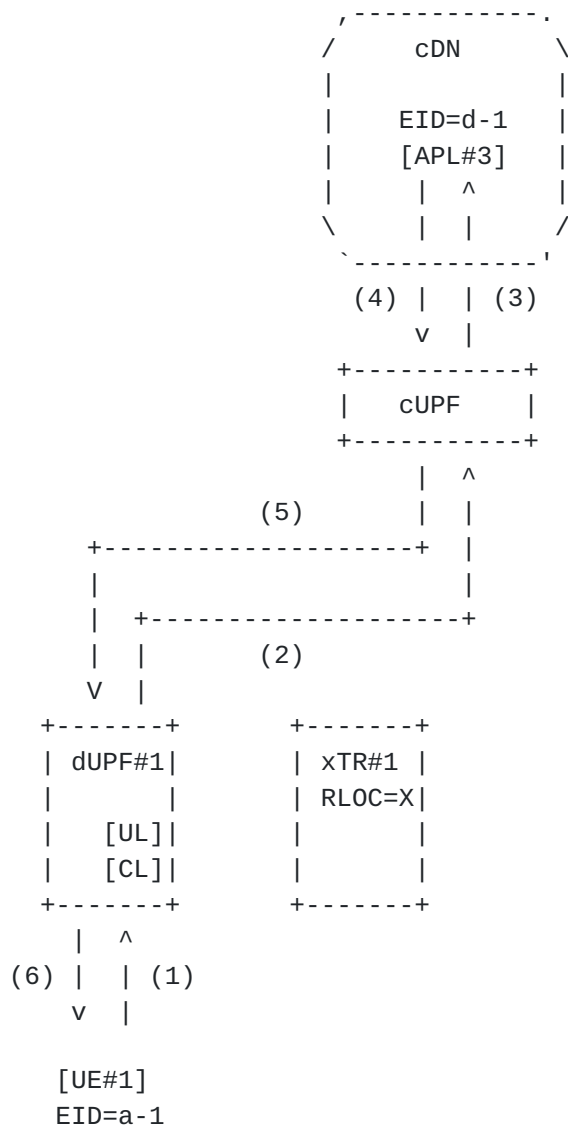


Figure 9: Procedure in Case A-5

- (0) Within this network, UEs are assigned their addresses from an address space [A]. These addresses are described as a-n (n=1,2,..). And applications in cDN are assigned their addresses from an address space [D]. These addresses are described as d-n (n=1,2,..). EID=a-1 and d-1 assigned to UE#1 and APL#3 which is located in cDN.

[Uplink Processes]

- (1) UE#1 sends packets to cDN with setting EID=d-1 as the destination IP address.

- (2) dUPF#1 monitors inner of received GTP-U packets. Since the destination IP address (EID=d-1) does not hit the filter of ULCL, dUPF#1 re-encapsulates the packet to another GTP-U connecting to cUPF and forwards to cUPF.
- (3) cUPF decapsulates GTP-U packets and forwards them to APL#3 in cDN depending on its own forwarding table.

[Downlink Processes]

- (4) APL#3 in cDN sends packets to UE#1 with setting EID=a-1 as the destination IP address.
- (5) cUPF encapsulates the packets received from APL#3 and forwards them to dUPF#1 depending on its own forwarding table.
- (6) dUPF re-encapsulates the packets to another GTP-U and forwards to UE#1.

Appendix B. Case Studies on Use of ILA

This Appendix describes detailed processes of the proposal approach with ILA mechanism in the following types of communications.

1. UE-to-UE Communication
2. UE-to-dDN Communication
3. UE-to-cDN/Internet Communication

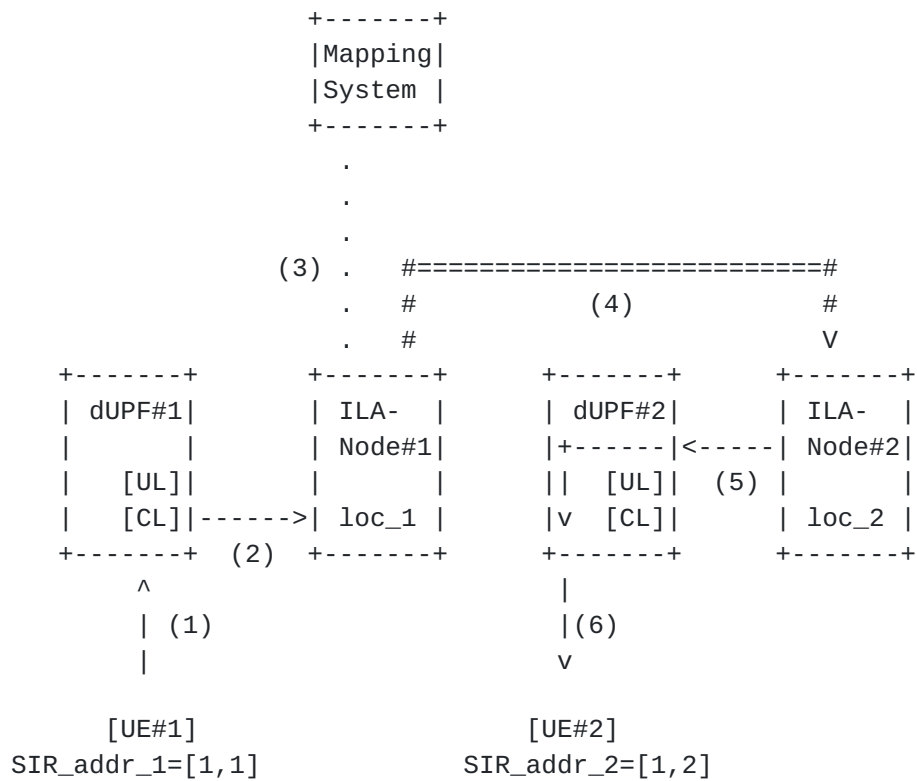
Each ILA node has ID-to-LOC mapping table. Mappings are propagated amongst ILA routers or hosts in a network using mapping propagation protocols.

In the following description of case studies, a mapping system, called ILA resolver in ILA terms, has the master of ID-to-LOC mapping database, and each ILA node obtains mappings from the mapping system. In some cases, each ILA node has an ID-to-LOC mapping database.

In ILA, an SIR address expressed by composition of SIR prefix and identifier is assigned to each UE or VM instance. An SIR prefix and an identifier are described SIR_prefix_n and id_m ($n=1,2,\dots$, $m=1,2,\dots$), and an SIR address is expressed as $SIR_addr_x = [n,m]$ ($x=1,2,\dots$) in the following description. Also, each ILA- Nodes are assigned unique Locators, which is a network prefix that routes to a host. Locators are described as loc_n ($n=1,2,\dots$).

B.1. UE-to-UE Communications

The overview of this communication type is described in A.1.

B.1.1. Case B-1: UEs allocated different dUPF

Legend: SIR_addr_x=[(SIR_Prefix), (Identifier)]

Figure 10: Procedure in Case B-1

- (0) Within this network, UEs are belonged to the same ILA domain, and the same SIR prefix is assigned to UEs. SIR_addr_1=[1,1] and SIR_addr_2=[1,2] are assigned to UE#1 and UE#2.
- (1) UE#1 sends packets to UE#2 with setting SIR_addr_2 as the destination IP address.
- (2) dUPF#1 monitors inner packet of received GTP-U packet and diverts it to ILA-Node#1 with decapsulation if the prefix of the destination address is SIR_prefix_1.

- (3) ILA-Node#1 updates own ID-to-LOC mapping table by interaction with the mapping system (if needed).
- (4) ILA-Node#1 obtains loc_2 as Locator of the ILA node#2 from the ID-to-LOC mapping table. ILA-Node#1 converts the prefixes of the source and destination addresses to loc_1 (Locator of id_1) and loc_2 (Locator of id_2). ILA-Node#1 sends the packet to the ILA-Node#2.
- (5) ILA-Node#2 receives the packet and converts the prefixes of the source and destination addresses to SIR_prefix_1, and then sends packets to dUPF#2.
- (6) dUPF#2 encapsulate packets with GTP-U header, and sends them to UE#2.

B.1.2. Case B-2: UEs allocated the same ILA node

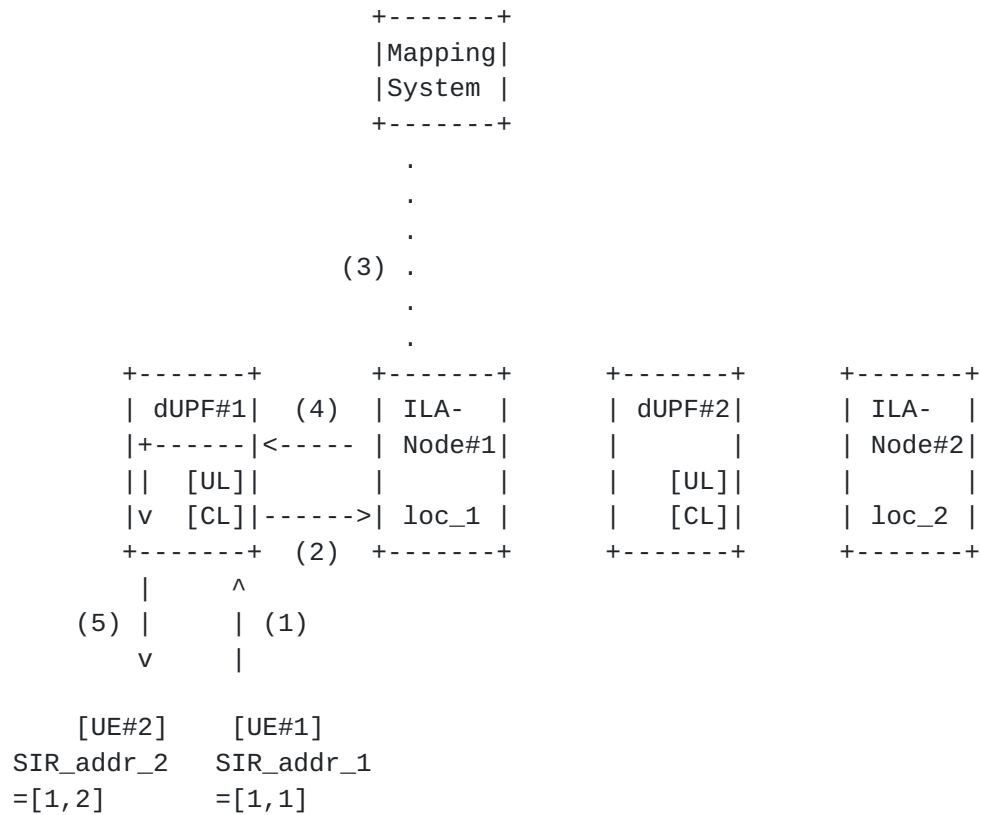


Figure 11: Procedure in Case B-2

- (0) Within this network, UEs are belonged to the same ILA domain, and the same SIR prefix is assigned to UEs. SIR_addr_1=[1,1] and SIR_addr_2=[1,2] are assigned to UE#1 and UE#2.
- (1) UE#1 sends packets to UE#2 with setting SIR_addr_2 as the destination IP address.
- (2) dUPF#1 monitors inner packet of received GTP-U packet and diverts it to ILA-Node#1 with decapsulation if the prefix of the destination address is SIR_prefix_1.
- (3) ILA-node#1 updates own ID-to-LOC mapping table by interaction with Mapping System (if needed).

(4) ILA-Node#1 obtains loc_1 as Locator of ILA node#2 from the ID-to-LOC mapping table. Since loc_1 indicates itself, ILA-Node#1 sends the packets back to dUPF#1.

(5) dUPF#1 encapsulate packets with GTP-U, and sends them to UE#2.

B.2. UE-to-dDN Communication

The overview of this communication type is described in A.2.

B.2.1. Case B-3: UE communicates with neighbor dDN

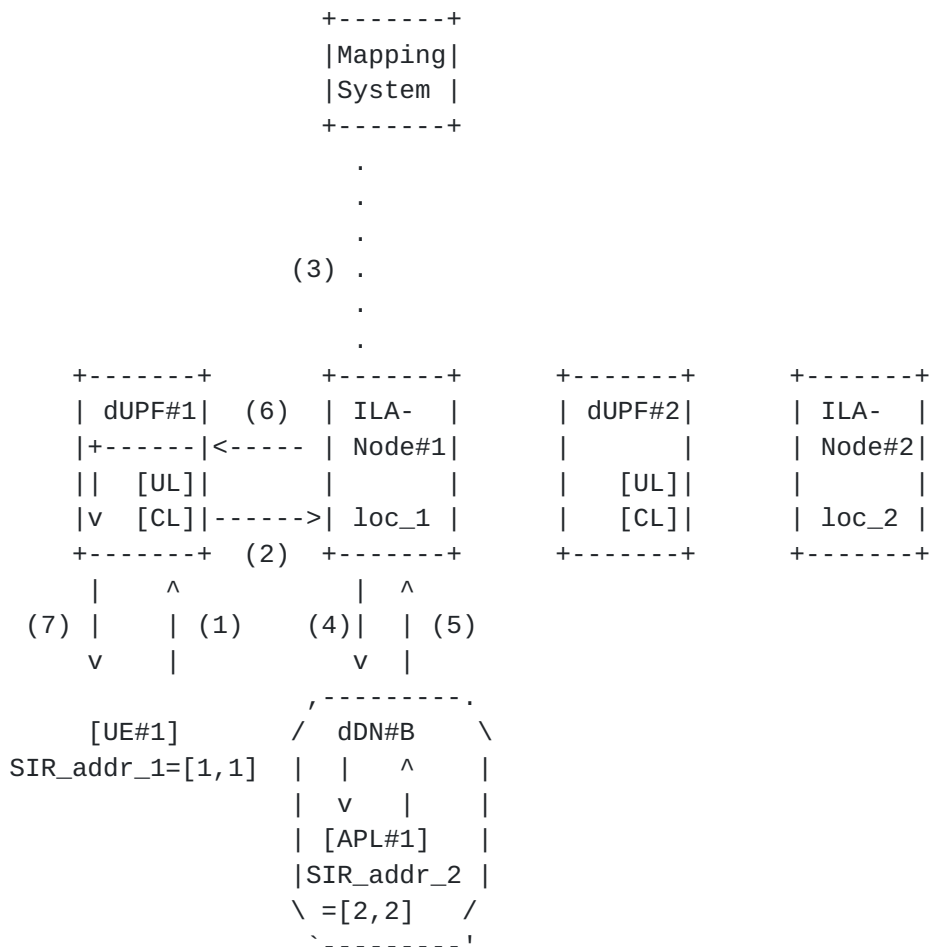


Figure 12: Procedure in Case B-3

- (0) Within this network, UEs are belonged to the same ILA domain, and the same SIR prefix (SIR_prefix_1) are assigned to UEs. Applications in dDN#B are belonged to different ILA domain. and different SIR prefix (SIR_prefix_2) is assigned to these applications. SIR_addr_1=[1,1] and SIR_addr_2=[2,2] are assigned to UE#1 and APL#1. APL#1 is located in dDN#B.

Uplink Processes

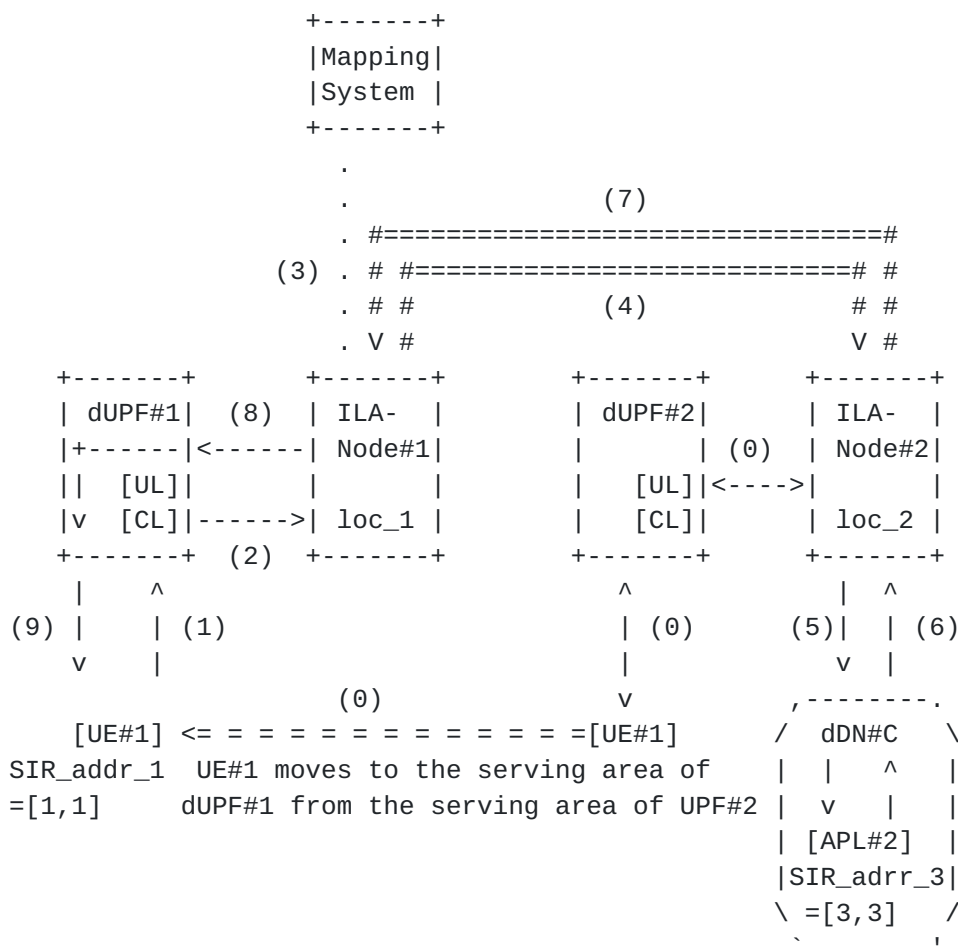
- (1) UE#1 sends packets to APL#1 with setting SIR_addr_2 as the destination IP address.

- (2) dUPF#1 monitors inner packet of received GTP-U packet and diverts it to ILA-Node#1 with decapsulation if the prefix of the destination address is SIR_prefix_2.
- (3) ILA-Node#1 updates own ID-to-LOC mapping table by interaction with a mapping system (if needed). Or ILA-Node#1 may update its own table by a Map-Notify message when an APL is deployed or deleted in dDB#B.
- (4) ILA-Node#1 obtains loc_1 as Locator of id_2 from the ID-to-LOC mapping table. Since loc_1 indicates itself, ILA-Node#1 sends the packets to the dDN#B.

Downlink Processes

- (5) APL#1 in dDN#B sends packets to UE#1 with setting SIR_address_1 as the destination IP address.
- (6) ILA-Node#1 obtains loc_1 as Locator of id_1 from the ID-to-LOC mapping table. Since loc=1 indicates itself, ILA-Node#1 sends packets to dUPF#1.
- (7) dUPF#2 encapsulates packets with GTP-U, and sends them to UE#1.

B.2.2. Case B-4: UE communicates with non-neighbor dDN



Legend: SIR_addr_x=[(SIR_Prefix), (Identifier)]

Figure 13: Procedure in Case B-4

- (0) Within this network, UEs are belonged to the same ILA domain, and the same SIR prefix (SIR_prefix_1) are assigned to UEs. Applications in dDN#C are belonged to different ILA domain. and different SIR prefix (SIR_prefix_3) is assigned to these applications. SIR_addr_1=[1,1] and SIR_addr_3=[3,3] are assigned to UE#1 and APL#2. APL#2 is located in dDN#C. UE#1 has moved to the serving area of dUPF#1 from the serving area of UPF#2 while communicating to APL#2.

Uplink Processes

- (1) UE#1 sends packets to APL#2 with setting SIR_addr_3 as the destination IP address.

- (2) dUPF#1 monitors inner packet of received GTP-U packet and diverts it to ILA-Node#1 with decapsulation if the prefix of the destination address is SIR_prefix_3.
- (3) ILA-Node#1 updates own ID-to-LOC mapping table by interaction with Mapping System (if needed).
- (4) ILA-Node#1 obtains loc_2 as Locator of id_3 from the ID-to-LOC mapping table. ILA-Node#1 converts the prefix of the source address to loc_1 (Locator of id_1), and the prefix of the destination address to loc_2 (Locator of id_3). ILA-Node#1 sends the packet to the ILA-Node#2.
- (5) ILA-Node#2 converts the prefix of the source address to SIR_prefix_1, and the prefix of the destination address to SIR_prefix_3, and then sends packets to dDN#C depending on its forwarding table.

Downlink Processes

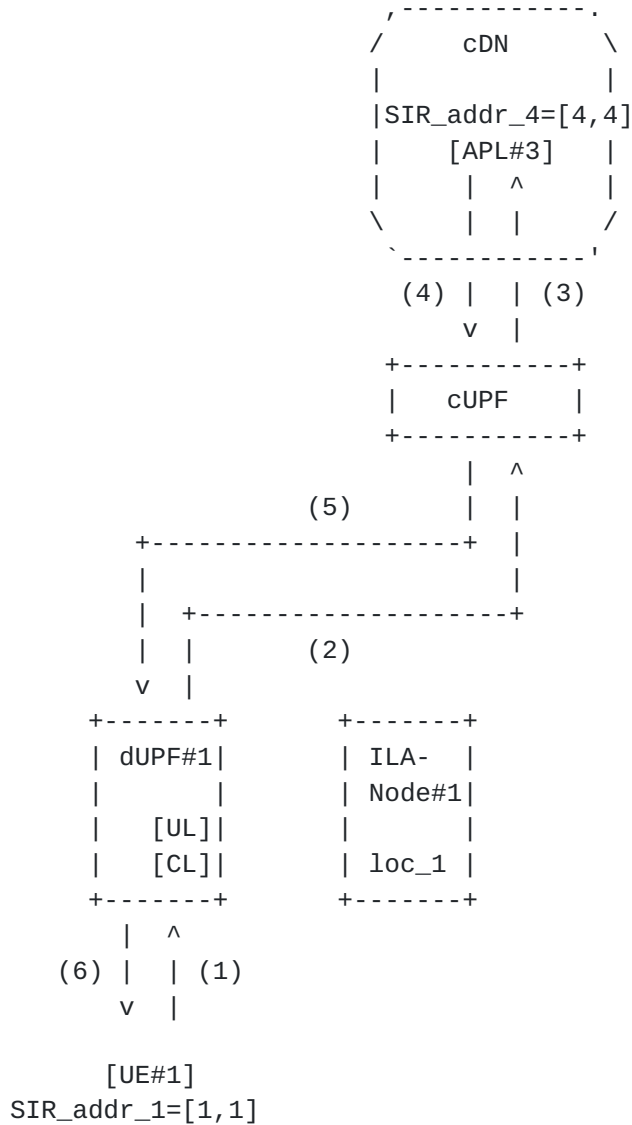
- (6) APL#2 sends packets to UE#1 with setting SIR_address_1 as the destination IP address.
- (7) ILA-Node#2 obtains loc_1 as Locator of id_1 from the ID-to-LOC mapping table. ILA-Node#2 converts the prefix of the source address to loc_2 (Locator of id_3), and the prefix of the destination address to loc_1 (Locator of id_1). ILA-Node#1 sends the packet to the ILA-Node#1.
- (8) ILA-Node#1 converts the prefix of the source address to SIR_prefix_3, and the prefix of the destination address to SIR_prefix_1, and then sends packets to d#UPF1 depending on its forwarding table.
- (9) dUPF#1 encapsulates the packets with GTP-U and sends packets to UE#1.

B.3. UE-to-cDN/Internet Communication

UE-to-cDN/Internet communication are basically achieved by GTP-U mechanism originally equipped in 3GPP 5GS architecture. ILA causes some limitation on IP addressing to UEs (e.g., all UEs in an ILA domain have the same SIR prefix), and thus some IP translation node such as NAT (Network Address Translation) may be required to enable UEs to access to external network. In this section, we describe processes of UE-to-cDN/Internet communication in the proposal architecture. In Internet communication, from aspect of privacy or routing with external network, SIR addresses assigned to UEs are

translated by NAT function deployed between dUPF and connection point.

B.3.1. Case B-5: Internet Communication



Legend: SIR_addr_x=[(SIR_Prefix), (Identifier)]

Figure 14: Procedure in Case B-5

- (0) Within this network, UEs are belonged to the same ILA domain, and the same SIR prefix (SIR_prefix_1) are assigned to UEs.

Applications in cDN are belonged to different ILA domain. and different SIR prefix (SIR_prefix_4) is assigned to these applications. SIR_addr_1=[1,1] and SIR_addr_4=[4,4] are assigned to UE#1 and APL#3. APL#3 is located in cDN.

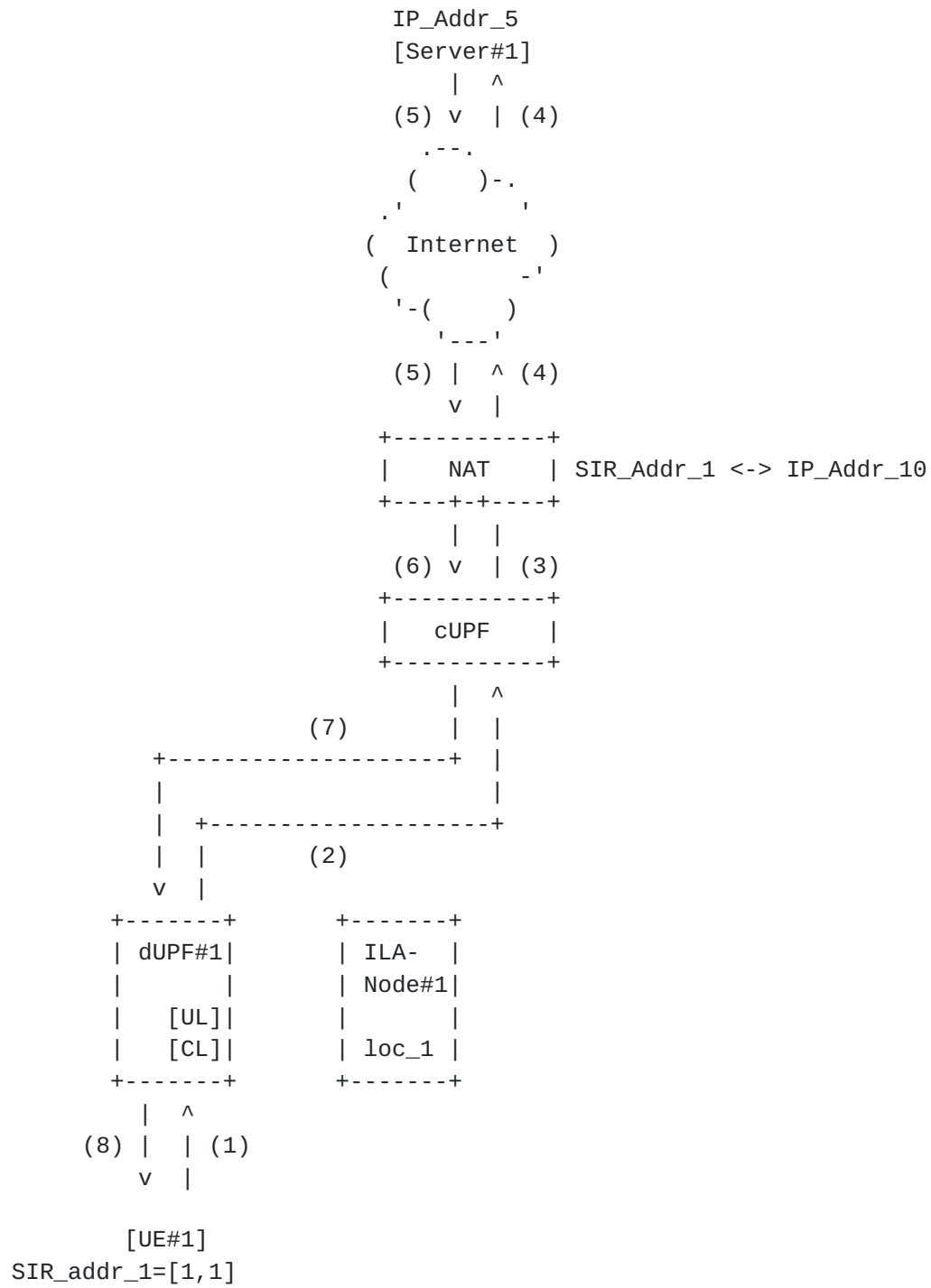
Uplink Processes

- (1) UE#1 sends packets to APL#3 with setting SIR_adrr_4 as the destination IP address.
- (2) dUPF#1 monitors inner of received GTP-U packets. Since the destination IP address (SIR_adrr_4) does not hit the filter of ULCL, dUPF#1 re-encapsulates the packet to another GTP-U connecting to cUPF and forwards to cUPF.
- (3) cUPF decapsulates GTP-U packets and forwards them to APL#3 in cDN depending on its own forwarding table.

Downlink Processes

- (4) APL#3 in cDN sends packets to UE#1 with setting SIR_addr_1 as the destination IP address.
- (5) cUPF encapsulates the packets received from APL#3 and forwards them to dUPF#1 with GTP-U encapsulation depending on its own forwarding table.
- (6) dUPF re-encapsulates the packets to another GTP-U and forwards to UE#1.

B.3.2. Case B-6: Internet Communication



Legend: SIR_addr_x=[(SIR_Prefix), (Identifier)]

Figure 15: Procedure in Case B-6

- (0) Within this network, UEs are belonged to the same ILA domain, and the same SIR prefix (SIR_prefix_1) are assigned to UEs. SIR_addr_1=[1,1] assigned to UE#1 and server#1 has IP_addr_5. UE#1 communicate with server#1 over Internet.

Uplink Processes

- (1) UE#1 sends packets to server#1 with setting IP_adrr_5 as the destination IP address.
- (2) dUPF#1 monitors inner of received GTP-U packets. Since the destination IP address (SIR_adrr_4) does not hit the filter of ULCL, dUPF#1 re-encapsulates the packet to another GTP-U connecting to cUPF and forwards to cUPF.
- (3) cUPF decapusalates GTP-U packets and forwards them to Internet depending on its own forwarding table.
- (4) NAT translates SIR_addr_1 of received packets to IP_addr_10 and the packets are forwarded to server#1 over Internet.

Downlink Processes

- (5) Server#1 sends packets to UE#1 with setting IP_addr_1 as the destination IP address.
- (6) NAT translates IP_addr_10 of received packets to SIR_addr_1, and packets are sent to cUPF.
- (7) cUPF encapsulates the packets with GTP-U and sends them to dUPF#1 depending on its own forwarding table.
- (8) dUPF re-encapsulates the packets to another GTP-U and forwards to UE#1.

Authors' Addresses

Shunsuke Homma
NTT, Corp.
3-9-11, Midori-cho
Musashino-shi, Tokyo 180-8585
Japan

Email: homma.shunsuke@lab.ntt.co.jp

Kenta Kawakami
NTT, Corp.
3-9-11, Midori-cho
Musashino-shi, Tokyo 180-8585
Japan

Email: kawakami.kenta@lab.ntt.co.jp

Arashmid Akhavain
Huawei Canada Research Centre
Canada

Email: arashmid.akhavain@huawei.com

