

5G コアネットワークにおける Locator/ID 分離と Locator 分割を用いたネットワークスライス構築

松枝 耕平[†] 落合 孝壮[†] 寺岡 文男^{††}

[†] 慶應義塾大学大学院理工学研究科 〒223-8522 神奈川県横浜市港北区日吉 3-14-1

^{††} 慶應義塾大学理工学部 〒223-8522 神奈川県横浜市港北区日吉 3-14-1

E-mail: †{baron,apollo}@inl.ics.keio.ac.jp, ††tera@keio.jp

あらまし 5G 時代の新しいサービスにおけるネットワーク要件は高度かつ多岐にわたる。5G では、単一の物理ネットワークに異なる特性の論理ネットワークであるネットワークスライスを構築し、サービスの種類ごとのネットワーク要件を満足する。ネットワークスライスの実現方式にはホップバイホップ方式とエッジオーバーレイ方式がある。ホップバイホップ方式では、SDN スイッチと SDN コントローラにより物理ネットワークを構築するため、柔軟なスライス構築が可能であるが導入が困難である。エッジオーバーレイ方式では、既存のインフラ資産を活用するが、トンネルによるスループット低下の問題がある。そこで本稿では、トンネル技術を使用しないネットワークスライス実現手法を提案する。IPv6 の Locator/ID 分離における Locator 部をスライスごとに分割することで、トンネルを撤廃したネットワークスライスの構築を実現する。Locator/ID 分離のプロトコルが適用可能であり、本研究室で提案しているトンネル技術を使わないモビリティ制御プロトコル MocLis を採用することができる。ゆえに、トンネル技術を完全に使用しないモバイルネットワークスライスを構築可能である。概念実証の結果、本提案方式の実現可能性が示された。
キーワード ネットワークスライシング, 5G, Loator/ID 分離

Construction of Network Slices by Using Locator/ID Separation and Locator Division in 5G Core Networks

Kohei MATSUEDA[†], Takamasa OCHIAI[†], and Fumio TERAOKA^{††}

[†] Graduate School of Science and Technology, Keio University Hiyoshi 3-14-1, Kohoku-ku, Yokohama, Kanagawa, 223-8522 Japan

^{††} Faculty of Science and Technology, Keio University Hiyoshi 3-14-1, Kohoku-ku, Yokohama, Kanagawa, 223-8522 Japan

E-mail: †{baron,apollo}@inl.ics.keio.ac.jp, ††tera@keio.jp

Abstract Network requirements for new services in the 5G era are more advanced and are highly diversified. In 5G, the network requirements for each service type are satisfied in each network slice. A network slice is a logical network of different characteristics constructed in a single physical network. There are two approach to create network slices. In Hop-by-Hop methods, it is hard to migrate from existing physical network infrastructure. In Edge Overlay methods, existing infrastructure are utilized, however tunneling causes the degradation of throughput. This paper proposes a network slice construction method by dividing the locator part for each slice in Locator/ID separation of IPv6. This approach eliminates tunneling. Locator/ID separation protocol e.g., MocLis is applicable for this. MocLis is a mobility support protocol that eliminates tunneling. Therefore, it is possible to construct a 5G mobile network that does not use tunneling. As a result of Proof of Concept, the feasibility of this approach is shown.

Key words Network Slicing, 5G, Locator/ID separation

1. はじめに

5Gでは、これまでのスマートフォンに加え、eMBB(enhanced Mobile Broadband:高速大容量通信),URLLC(Ultra-Reliable and Low Latency Communications:超高信頼低遅延通信),mMTC(massive Machine-Type Communications:大量端末通信)などの特性を活かした、様々な通信サービスが求められる[1]。このような5G時代の通信サービスにおける要求事項は高度かつ多岐にわたる。特性が異なる複数の通信サービスの要求事項を同時に満足することは困難である。5Gでは、単一の物理ネットワークに異なる特性の論理ネットワーク、すなわちネットワークスライスを構築し、スライスごとに特定の通信サービスを経済的に実現することを目指している。例えば、単一の5Gの物理ネットワーク上に4つのネットワークスライスを構築し、1つ目のスライスはベストエフォート通信、2つ目のスライスはeMBB用、3つ目のスライスはURLLC用、4つ目のスライスはmMTC用とすることができる。ネットワークスライスごとに、使用する資源は他のスライスから隔離される。

ネットワークスライスの実現方式には、ホップバイホップ方式とエッジオーバーレイ方式がある。ホップバイホップ方式では、パケット処理をSDN(Software-Defined Networking)スイッチに設定してスライスを構築する。SDNスイッチとSDNコントローラにより物理ネットワークを構築するため、柔軟なスライス構築が可能であるが、既存のシステムとのマイグレーションが困難である。2つ目のエッジオーバーレイ方式では、L3上にL2リンクを確立してスライスを構築する。既存のインフラ資産を活用するが、トンネリング処理によるスループット低下の問題がある。

そこで本稿では、トンネリング処理を使用しないネットワークスライス実現手法を提案する。本提案方式は、IPv6のLocator/ID分離におけるLocatorをスライスごとに分割することでスライスを構築し、通常の経路制御プロトコルにより経路制御する。Locator/ID分離のプロトコルが適用可能であり、本研究室で提案しているトンネリングを撤廃した5Gシステム用モビリティ制御プロトコルMocLis[2]を採用することで、スライス構築およびスライス上での通信の両方でトンネリングを撤廃した5Gネットワークを構築可能である。

2. ネットワークスライス構築手法

2.1 ホップバイホップ方式

ホップバイホップ方式では、図1のように、SDNスイッチとSDNコントローラによって物理ネットワークを構築する。SDNコントローラは複数台のSDNスイッチを制御し、SDNスイッチに対するパケット処理を指示する。SDNスイッチはSDNコントローラの指示をフローテーブルに保持する。フローテーブルには、パケットマッチングに使用するフィールドとその値の定義、マッチしたパケットに対する処理(指定したポートへの中継、廃棄、フィールドの書き換えなど)の定義、マッチしたパケット数などを保持する。SDNスイッチに対して、スライスごとのパケットマッチングおよびマッチしたパケットに

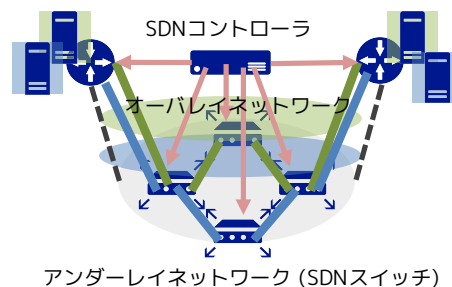


図1 ホップバイホップ方式

対する中継先を設定することで、ネットワークスライスを構築する。SDNコントローラとSDNスイッチ間の通信を規定するプロトコルとして、OpenFlow[3]が提案されている。ホップバイホップ方式におけるスライス間でのネットワーク資源隔離は、SDNスイッチに実装された資源隔離機能に依存する。

ホップバイホップ方式の課題を以下に挙げる。

- 物理ネットワークをSDN対応機器で構築する必要があり、導入が困難である。
- SDNコントローラがSDNスイッチを集中制御するため、SDNコントローラが単一障害点となる。
- SDNスイッチはフローテーブルに従って動作するため、ネットワーク障害が発生した場合の障害原因の切り分けが困難である。

2.2 エッジオーバーレイ方式

エッジオーバーレイ方式は、ネットワーク層(L3)の上に、トンネリングによって仮想的なデータリンク層リンク(L2)を確立してネットワークスライスを構築する。トンネリングプロトコルとして、VXLAN(Virtual eXtensible Local Area Network)[4]やNVGRE(Network Virtualization using Generic Routing Encapsulation)[5]がある。図2のように、トンネリングの端点はトンネリング処理機能を持つルータで構築する必要があるが、その他のネットワーク機器は既存のものを使用することができる。エッジオーバーレイ方式における資源の隔離は、MPLS(Multi-Protocol Label Switching)[6]により実現可能である。

VXLANの利用したエッジオーバーレイ式の課題を以下に挙げる。

- VXLANのカプセル化のヘッダオーバーヘッドは、IPv4の場合54バイト、IPv6の場合74バイトと非常に大きい。
- ヘッダオーバーヘッドが大きいため、VXLANの入口ノードのIP層においてロングパケットが発生することがある。
 - ロングパケットはパケットのフラグメンテーションとリアセンブリを発生させる。
 - IPv6ではIPv6 Path MTU Discoveryによりフラグメンテーションが発生しないが、オーバーレイネットワーク上の仮想ノードはアンダーレイネットワークで発生するカプセル化を認識できず、ロングパケットを解消できない[7]。
- カプセル化されたパケットはUDPで伝搬されるため、TSO(TCP Segmentation Offload)やLRO(Large Receive Offload)などのTCPのオフローディング機能を使用できず、

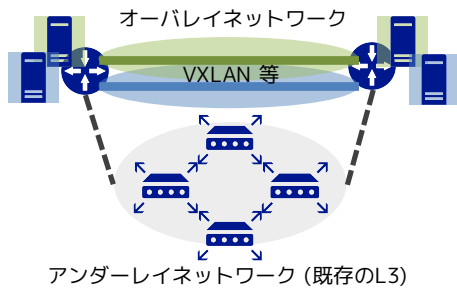


図 2 エッジオーバレイ方式

トンネリング処理なしの場合と比べスループットが大幅に低下する [8] .

3. トンネリング処理を利用しないモビリティプロトコル

モビリティサポートは、User Equipment (UE) がネットワークに接続している間の IP アドレスの継続性として提供される。IP アドレスは、識別子とロケータの両方の意味を持つ。モバイルネットワークでは、UE がある基地局から別の基地局に接続先を変更 (ハンドオーバ) する場合に、IP アドレスの 2 つの関係を維持できない。この問題の解決策が長年に渡り多数提案されている [9] . LTE-Advanced では、モビリティ制御プロトコルとして Proxy Mobile IPv6 (PMIPv6) [10] が採用されている [11] . LTE-Advanced のコアネットワーク (EPC: Evolved Packet Core) は、UE の IP アドレス宛のパケットに対し、UE が現在接続している基地局の IP アドレスとトンネルを確立している EPC 内のゲートウェイを使用する。UE がハンドオーバすると、パケットは新しい基地局にトンネリングされる。

トンネリング処理のオーバーヘッドの問題を解決するために、5G に向けて、トンネリング処理を使用しないモビリティプロトコルがいくつか提案されている。

3.1 SimpleCore

SimpleCore [12] は、ビデオストリーミングやウェブページ閲覧に基づき、5G におけるデフォルトサービスはコネクションレス、ベストエフォート、モビリティサポートなしで動作可能であると想定し、コアネットワーク内のノード数を削減しプロトコルスタックを単純化したコアネットワーク設計である。SimpleCore では、基地局にあらかじめ IP アドレスブロックが割り当てられる。UE がモバイルネットワークに接続すると、基地局から、現在位置に基づく IP アドレスが割り当てられる。ゆえに、割り当てられる IP アドレスは基地局からインターネットまでルーティングが可能である。UE がハンドオーバすると、UE に対し新たな基地局位置に基づく新しい IP アドレスが割り当てられる。UE が IP アドレスの継続性を要求する場合は、モビリティサポートなしのデフォルトの IP アドレスの他に、2 つ目の IP アドレスを UE に割り当てるが、その IP アドレスに対しては、従来のモビリティサポートプロトコルと同様にトンネリング処理を必要とする。

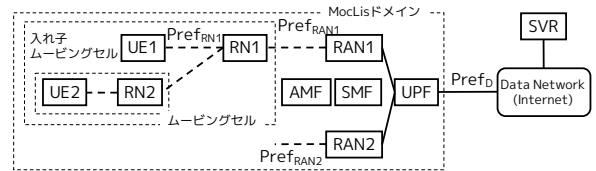


図 3 MocLis のシステム構成例

3.2 MocLis (Moving Cell by Locator/ID Separation)

MocLis (Moving Cell by Locator/ID Separation) [2] は、トンネリング処理を利用せず、かつ移動式のセルであるムービングセルをサポート可能なモビリティサポートプロトコルである。Locator/ID 分離に基づき、IPv6 アドレスを位置と識別子の 2 つの情報に分離して利用することで、シグナリングを単純化し、データ転送におけるヘッダオーバーヘッドをなくしている。

図 3 に MocLis のシステム構成例を示す。MocLis ドメインは、LocLis が動作する範囲であり、例えば一つの 5G システムである。UPF (User Plane Function) は、MocLis ドメインとインターネットの境界にあるルータである。RAN (Radio Access Network) は固定基地局、RN (Relay Node) は移動式基地局である。AMF (Access and Mobility Function) は、ノードの識別子とロケータの関係を管理する。図 3 において、ゲートウェイルータの上流リンクは $Pref_D$ という IPv6 プレフィックス空間を持つ。RAN1 と RAN2 の下流リンクはそれぞれ $Pref_{RAN1}$, $Pref_{RAN2}$ という IPv6 プレフィックス空間を持つ。RN1 は UE と RN2 にムービングセルを提供し、RN2 は UE2 に入れ子ムービングセルを提供する。RN1 の下流リンクは $Pref_{RN1}$ を持つ。SVR (Server) はインターネット上に存在するノードで、MocLis ドメイン内の UE1 の通信相手である。

MocLis では、Locator/ID 分離に基づき、IPv6 アドレスを上位 64 ビットの Locator 部と下位 64 ビットの ID 部に分ける。図 3 において、 $Pref_D$ はインターネット上で一意である。UE1 の ID を ID_{UE1} と表すと、SVR は UE1 の IPv6 アドレスを $[Pref_D:ID_{UE1}]$ と認識する。図 4 に MocLis における UE1 と SVR の通信例を示す。SVR が $[Pref_D:ID_{UE1}]$ 宛のパケットを送信すると、インターネット上のルーティングに基づき、ゲートウェイルータに到達する。その後 MocLis ドメインにて、Locator 部を書き換えながら、終点アドレスが $[Pref_{RN1}:ID_{UE1}]$ になった IPv6 パケットが UE1 に到着する。UE1 のネットワーク層にて、受信パケットの終点アドレスの Locator 部を $Pref_D$ 書き戻すことにより、UE1 上のアプリケーションは SVR とエンドツーエンドの通信ができていると認識する。IPv6 アドレスの Locator 部の書き換えは、トンネリング処理より軽量である。

4. 5G コアネットワーク

3GPP (3rd Generation Partnership Project) [13] では、5G コアネットワークを 2018 年 6 月までに 3GPP Release15 として策定することが計画されている。

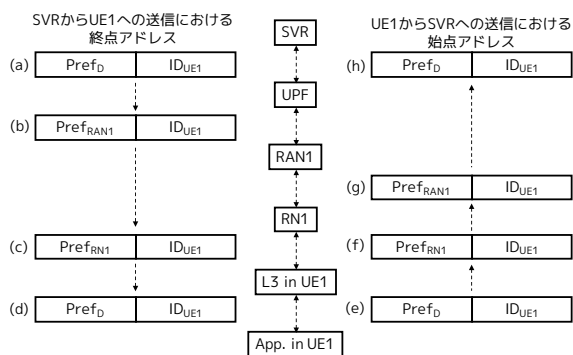


図 4 SVR と UE1 間の Locator の書き換え

4.1 ノード

図 5 に、Network Function (NF) と NF 間インタフェースからなる 5G (NextGen) のリファレンスアーキテクチャ [14] を示す。User Equipments (UE) は Radio Access Network (RAN) または Access Network (AN) および Access and Mobility Function (AMF) に接続している。RAN は new RAT (Radio Access Technology) を利用する基地局である。AN は非 3GPP アクセスを利用する基地局であり、例えば Wi-Fi のアクセスポイントである。5G のコアネットワークは次の NF で構成される。(1) AMF、(2) Session Management Function (SMF)、(3) Policy Control Function (PCF)、(4) Application Function (AF)、(5) Authentication Server Function (AUSF)、(6) User Data Management (UDM)、(7) User Plane Function (UPF) である。

3GPP では、通信の確立などの制御信号通信のための Control Plane (C-Plane) とユーザデータ通信のための User Plane (U-Plane) を分離して処理する C/U 分離の考え方を適用した CUP (Control and User Plane Separation of EPC nodes) という、EPC アーキテクチャの拡張が標準化されている [15]。そして、カバー範囲の狭い小セルを数多く設置するとハンドオーバが頻繁に発生しスループットの低下が懸念されるため、広範囲をカバーするマクロセルで C-Plane を利用し、高スループットの小セルで U-Plane を利用するファントムセルが提案されている [16]。こうした背景から、5G では C/U 分離を考慮したアーキテクチャが採用される。C-Plane には (1) - (6) が属し、U-Plane には (7) が属する。

AMF は UE の認証、認可、モビリティ管理などを提供する。SMF は UE のセッション管理と IP アドレスの割当、データ転送のための UPF の選択と制御を担当する。UE が複数のセッションを確立する場合、各セッションを独立して管理し、セッションごとに異なるファンクションを利用するために、異なる SMF を各セッションに割当可能である。5G コアネットワークでは、UE に関わる管理は 1 つの AMF で行い、トラフィックは個別のネットワークスライスごとの SMF で扱う。AMF は SMF を制御する。AF は QoS (Quality of Service) をサポートするために、ポリシー制御を提供する PCF に対し、パケットフローに関する情報を提供する。この情報に基づき、PCF は、AMF と SMF を適切に動作させるための、モビリティおよ

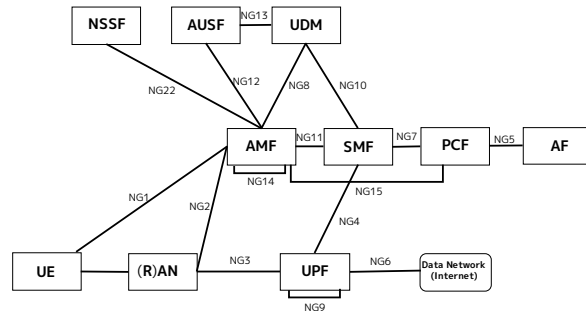


図 5 5G リファレンスアーキテクチャ [14]

びセッション管理に関するポリシーを決定する。AUSF は UE の認証のためのデータを保存し、UDM は UE の契約情報を保存する。5G コアネットワークに含まれないが、Data Network がインターネットへのアクセスを提供する。UPF は EPC に CUPS における U-Plane 処理機能を提供する。

3GPP では、AMF や NSSF などの全セッション (全ネットワークスライス) で共通にサポートする NF を、Common Control Network Functions (CCNF) として定義している。

4.2 インタフェース

5G コアネットワークではインタフェースとして、NG (Next Generation) 1 が UE と AMF の間のシグナリングのために定義されている。AN と AMF の間および AN と UPF の間のインタフェースは NG2, NG3 として定義されている。AN と SMF の間のインタフェースは存在しないが、AMF と SMF の間に NG11 が定義されている。NG4 は SMF と UPF の間で定義されており、SMF が UPF の状態を設定し、UPF が SMF に状態を通知するのに利用される。NG9 は UPF 間で、NG14 は異なる AMF 間の通信のために定義されている。NG15 と NG7 は、PCF が SMF と AMF にポリシーを適用するために定義されている。NG12 は SMF が UE の認証のために定義されている。NG8 と NG10 は、AMF と SMF が、UE の契約情報を UDM と通信するために定義されている。NG22 は AMF と NSSF の間で定義され、AMF が UE 用の適切な SMF を選択するのに利用される。

5. 提案スライス構築法

本提案の適用範囲は、5G ネットワークのような 1 つのネットワークオペレータが運用するネットワーク内であり、ネットワーク内では IPv6 が適用されているものとする。本提案方式におけるネットワーク資源隔離は、アンダーレイネットワークを構成するハードウェアの機能により実現することを想定している。

5.1 Locator/ID 分離に基づく IPv6 アドレス形式

図 6 に Locator/ID 分離に基づく IPv6 アドレス形式を示す。Locator/ID 分離とは、16 バイトの IPv6 アドレスを上位 8 バイト (第 0 バイト-第 7 バイト) と下位 8 バイト (第 8 バイト-第 15 バイト) に分け、上位 8 バイトをノードが接続するサブネット番号 (Locator)、下位 8 バイトをノードの識別子 (ID) として解釈するという考え方である [17]。

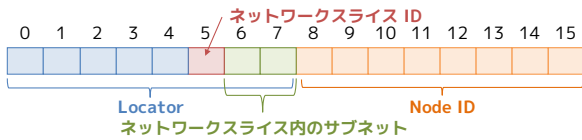


図 6 Locator/ID 分離に基づく IPv6 アドレス形式

Locator の第 5 バイトをスライス ID, Locator の第 6-7 バイトをスライス内のサブネット番号を表すものと定義する。すなわち, VXLAN におけるスライス ID である VNI (Virtual Network Identifier) の代わりに, 提案方式では, Locator の第 6-7 バイトをスライス ID として利用する。したがって, スライス番号は 0-127 までの 128 個となり, 各スライスには 65,536 個のサブネットを定義することができる。ネットワークスライスはサービスの特性ごとに分類されることを想定しており, 2 バイトあれば必要十分である。なお, 上記の定義はネットワークオペレータが/40 の IPv6 アドレスプレフィックスをネットワークに割り当てることを想定した一例であり, スライス ID を示すフィールドやサブネット番号を示すフィールドのビット数を変更してもよい。

Loc₀₁ を IPv6 アドレスの第 0 バイトと第 1 バイトを表す 2 バイトの値, Loc₂₃ を IPv6 アドレスの第 2 バイトと第 3 バイトを表す 2 バイトの値, Loc₄ を IPv6 アドレスの第 4 バイトを表す 1 バイトの値, Slice をスライス ID を表す 1 バイトの値, Subnet をサブネット番号を表す 2 バイトの値, ID を IPv6 アドレスの第 8 バイトから第 15 バイト (ID 部) を表す 8 バイトの値とすると, 本提案における IPv6 アドレスは以下のように表される。

Loc₀₁:Loc₂₃:Loc₄Slice:Subnet:ID

5.2 ネットワークスライス構築例

図 7 に, 物理ネットワーク上に 2 つのネットワークスライスを構築した例を示す。ネットワークスライス 1, 2 が使用する IPv6 プレフィックスはそれぞれ次のとおりである。

Loc₀₁:Loc₂₃:Loc₄01::/48

Loc₀₁:Loc₂₃:Loc₄02::/48

図 7 において, ネットワークスライス上のノードは, 物理ネットワークの物理ノード上の仮想ノードとして存在する。構築例では仮想ノードのみでネットワークスライスを構築しているが, ネットワークスライス上のノードは仮想ノードまたは物理ノードである。仮想ノードは, 図 7 の縦の点線で結ばれる物理ネットワーク上の物理ノードで動作する。ネットワークスライス上のノードは, ネットワークスライスごとに異なる ID (IPv6 アドレスの下部 8 バイト) を持つ。例えば, ネットワークスライス 1, 2 における UPF1 の仮想ノード (UPF1A, UPF1B) の ID はそれぞれ ID^{UPF1A}, ID^{UPF1B} という 8 バイトの値である。

ID^{UPF1A}

ID^{UPF1B}

ネットワークスライスを構成するためのヘッダオーバーヘッドは生じない。上記のようにネットワークスライスを構築するため, 各物理ノード・仮想ノードは自ノードが属するネットワー

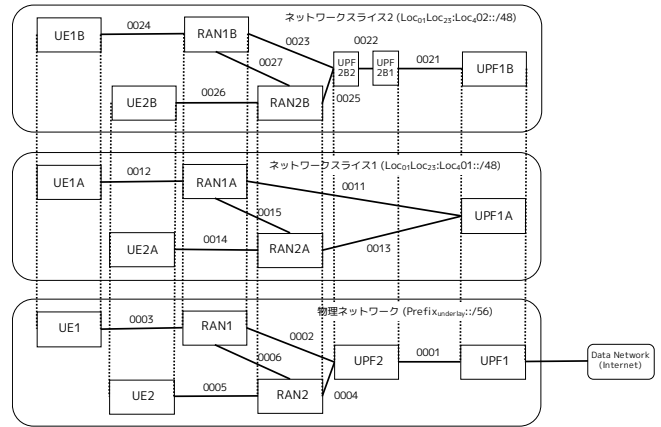


図 7 ネットワークスライス構築例

表 1 UPF1 のスライス 1 上の仮想ノード UPF1A が保持する経路表

宛先	ネクストホップ
Loc ₀₁ :Loc ₂₃ :Loc ₄ 01:0012::/64	IP ₁ ^{UPF2A}
Loc ₀₁ :Loc ₂₃ :Loc ₄ 01:0013::/64	IP ₁ ^{UPF2A}
Loc ₀₁ :Loc ₂₃ :Loc ₄ 01:0014::/64	IP ₁ ^{UPF2A}
Loc ₀₁ :Loc ₂₃ :Loc ₄ 01:0015::/64	IP ₁ ^{UPF2A}
Loc ₀₁ :Loc ₂₃ :Loc ₄ 01:0016::/64	IP ₁ ^{UPF2A}
default	IP ^{GW}

表 2 RAN1 のスライス 1 上の仮想ノード RAN1A が保持する経路表

宛先	ネクストホップ
Loc ₀₁ :Loc ₂₃ :Loc ₄ 01:0013::/64	IP ₁ ^{RAN2A}
Loc ₀₁ :Loc ₂₃ :Loc ₄ 01:0014::/64	IP ₁ ^{RAN2A}
default	IP ₁ ^{UPF1A}

表 3 RAN2 のスライス 1 上の仮想ノード RAN2A が保持する経路表

宛先	ネクストホップ
Loc ₀₁ :Loc ₂₃ :Loc ₄ 01:0011::/64	IP ₁ ^{UPF1A}
Loc ₀₁ :Loc ₂₃ :Loc ₄ 01:0012::/64	IP ₁ ^{RAN1A}
default	IP ₁ ^{UPF1A}

表 4 UE1 のスライス 1 上の仮想ノード UE1A が保持する経路表

宛先	ネクストホップ
default	IP ₁ ^{RAN1A}

表 5 UE2 のスライス 1 上の仮想ノード UE2A が保持する経路表

宛先	ネクストホップ
default	IP ₁ ^{RAN2A}

クスライス内での経路表を持つ。表 1-5 にネットワークスライス 1 上の仮想ノードの経路表をそれぞれ示す。IP^{NodeName}_{SliceNum} は, NodeName というノードが SliceNum というネットワークスライスにおいて持つ IPv6 アドレスを表す。IP^{GW} は, インターネット側で UPF1 と接続しているゲートウェイルータの IPv6 アドレスを表す。

5.3 パケットの転送例

UE1 上のウェブブラウザアプリケーションがインターネット上のサーバである SVR で動作するウェブサーバにアクセスする場合を想定する。ウェブ閲覧アプリケーションに関わるパ

ケットは、図 7 のネットワークスライス 1 を利用する。UE1 から SVR に送信する IPv6 パケットの始点 IPv6 アドレスと終点 IPv6 アドレスは次のとおりである。

始点： IP_1^{UE1A} (= Loc01:Loc23:Loc401:0012:ID)

終点： IP^{SVR}

UE1 のネットワークスライス 1 での仮想ノード UE1A は、自ノードの経路表(表 4)の 1 つ目のエントリを参照し、RAN1A へパケットを転送する。このパケットは、アンダーレイの物理ノード UE1 が持つネットワークスライス 1 のノードに対する経路表のエントリにしたがって RAN1 へ転送される。ゆえに、アンダーレイの物理ノードの経路表は、ネットワークスライス数に応じてエントリが増える。RAN1 へ転送されたパケットは、自ノードのネットワークスライス 1 のノードに対する経路表のエントリにしたがって、RAN1A へ転送される。RAN1A がこのパケットを受信すると、自ノードの経路表(表 4)の 3 つ目のエントリを参照し、UPF1A へパケットを転送する。このパケットは、アンダーレイの物理ノード RAN1 のネットワークスライス 1 に対する経路表のエントリにしたがって UPF2 へ転送される。UPF2 がこのパケットを受信すると、ネットワークスライス 1 のノードに対する自身の経路表にしたがって UPF1 へ転送される。UPF1 がこのパケットを受信すると、自ノードのネットワークスライス 1 のノードに対する経路表のエントリにしたがって、UPF1A へパケットを転送する。UPF1A が転送されたパケットを受信すると、自ノードの経路表(表 1)の 6 つ目のエントリにしたがって、アンダーレイの UPF1 の経路制御に基づき、インターネット側のゲートウェイにこのパケットを転送する。実際には、インターネット側では通常の経路制御によりこのパケットは SVR に到達する。

終点と始点を反対にした場合も、通常の経路制御で UPF1 までパケットが到達すると、UPF1 は終点アドレスの第 5 バイトの値が 1 であることから、このパケットがネットワークスライス 1 に属するノード宛であることを認識する。UPF1A がこのパケットを受信すると、前述と同様の経路制御により、UE1A までパケットが到達する。

5.4 C/U/M 分離のスライス構成

5G の設計原則に基づき、本提案方式による 5G システムにおけるスライス構築でも C/U 分離を採用する。C-Plane に相当する Control Slice (C-Slice) と U-Plane に相当する User Slice (U-Slice) をサービスの種類ごとに対で使用する。すなわち、特性の異なるサービスごとに、異なる C-/U-Slice の対を利用する。そして、C-Slice と U-Slice の管理および C-Slice の共通機能を提供するスライスとして、Management Slice (M-Slice) を導入し、C/U/M 分離の 3 層のスライス構成を適用する。M-Slice が C-Slice を制御し、C-Slice が U-Slice を制御する。C/U/M 分離のスライス構成の概念を図 8 に示す。5G のリファレンスアーキテクチャに 3 層のスライス構成を適用させたシステムを図 9 に示す。

M-Slice と C-Slice の間のインタフェースは M-Slice に属し、C-Slice と U-Slice の間のインタフェースは C-Slice に属する。図 9 において、PCF (Policy Control Function) および AF

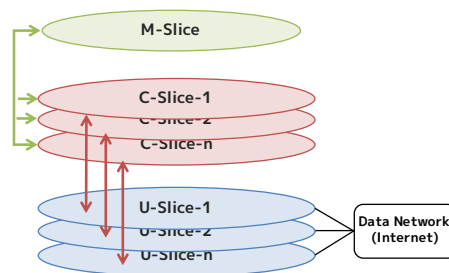


図 8 C/U/M 分離のスライス構成

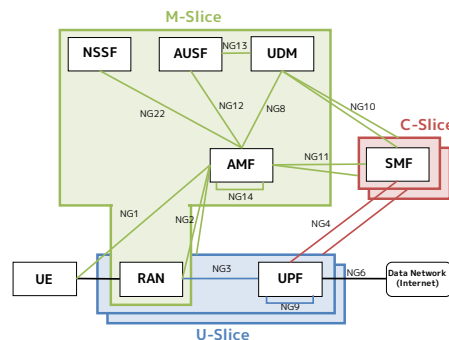


図 9 5G システムへの C/U/M 分離のスライス構成の適用

(Application Function) は省略している。

5.4.1 Management Slice (M-Slice)

Management Slice (M-Slice) は、UE に対する IP アドレスの管理や認証・認可、加入者情報の管理を行い、C-Slice を制御するネットワークスライスである。M-Slice に属するノードは、NSSF, AUSF, AMF, UDM, RAN である。M-Slice はネットワーク事業者につき 1 つである。

5.4.2 Control Slice (C-Slice)

Control-Slice (C-Slice) は、UE (User Equipment) とコアネットワークの間の制御メッセージの通信スライスであり、1:1 の対となる U-Slice の UPF を制御する。C-Slice に属するノードは SMF である。

5.4.3 User Slice (U-Slice)

User Slice (U-Slice) は、UE のデータ通信スライスである。C-Slice と 1:1 の対である。U-Slice に属するノードは UPF と RAN である。

5.5 UE の接続手順

UE の 5G ネットワークへの接続手順を図 10 に示す。UE が起動すると、基地局との間で Router Solicitation (RS) メッセージと Router Advertisement (RA) メッセージを交換することで、UE が M-Slice で利用する IP アドレスを取得する。この IP アドレスを利用し、UE は NSSF へ Attach Request を送信する。NSSF は AUSF に問い合わせ、UE が認証されているか確認する。認証された UE からの Attach Request の場合、NSSF は Attach Request に含まれる UE の種類やサービスのタイプの情報から、適切な Network Slice Instance (NSI) として、C-/U-Slice のペアを選択する。そして、AMF へ Attach Request を転送する。AMF は、Attach Request に含まれる C-/U-Slice の情報を元に、UE がそのネットワークスライス

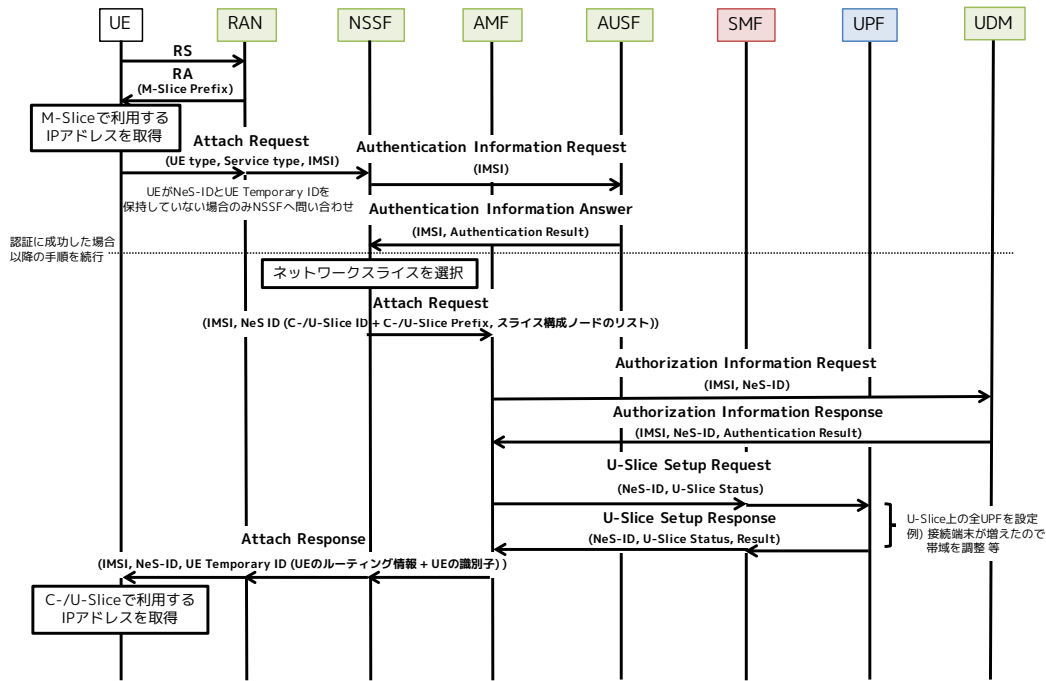


図 10 UE のスライス接続手順

利用できるよう認可する．認可されている UE の場合，AMF は選択された C-Slice 上の SMF に対して，U-Slice の状態を設定するよう，U-Slice Setup Request を送信する．このメッセージを受信した SMF は，SMF が管理する U-Slice 上の全 UPF に対して，パラメータの設定等を行う．それが完了すると，SMF は AMF に U-Slice Setup Response を送信する．AMF が U-Slice Setup Response を受信すると，UE に対して一意の UE Temporary ID および，C-/U-Slice の情報を含む Attach Response を UE に送信する．UE が Attach Resonse を受信すると，C-/U-Slice の情報から自身の IP アドレスを設定し，UE Temporary ID に含まれる UE の経路制御情報に基づき，自身の経路を設定する．これにより UE が適切な C-/U-Slice へ接続を完了する．

6. 概念実証と評価

6.1 コンテナ型仮想化技術によるネットワークスライス構築の検証

コンテナは，仮想マシンより軽量な隔離された OS (Operating System) の環境を持ち，その中でアプリケーションを動作させることができる．例えば OS として Linux [18] があり，コンテナとして Docker [19] や LXC [20] がある．コンテナはホスト OS とカーネルなどのリソースを共有するため，コンテナ内で動作するアプリケーションのオーバーヘッドは仮想マシンに比べ小さい．

物理マシン (ハイパーバイザ: VMWare ESXi 4.1.0, CPU: AMD Opteron 2439 SE 2.8 GHz, メモリ: 64 GB) 上で図 11 の 5 台の仮想マシン (CPU 数: 1, メモリ: 1GB, OS: Ubuntu 16.04.3 LTS, カーネル: Linux Kernel 4.4.0) をアンダーレイネットワークを構築するノードとして動作させた．そのうち 3

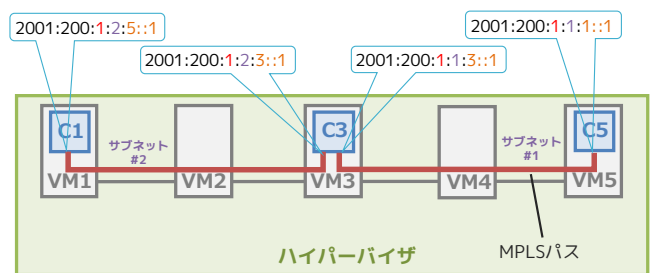


図 11 ネットワークスライス構築の検証

台の仮想マシン上で，ネットワークスライスを構築する仮想ノードとして Docker コンテナ (OS: Ubuntu 16.04.3 LTS) をそれぞれ 1 台動作させた．

ネットワークスライスでは仮想ノード間の資源ネットワーク資源隔離が必要となる．今回は，仮想ノード間のネットワーク資源隔離のために，アンダーレイネットワークを構築する仮想マシンにおいて，mpls_router カーネルモジュールにより MPLS を利用した．パケットの送信元の仮想ノードで MPLS ヘッダを付与し，MPLS ラベルをスワップしながら次の仮想ノードまでパケットを転送する．ネットワークスライス上のパケット転送で途中に通過する仮想ノードにて，一度 MPLS ヘッダを外すことで仮想ノードにパケットを引き上げる．

3 つのコンテナの間で MPLS パスを構築し，図 11 のネットワークを構築し，ネットワークスライス内の仮想ノード間 (C1-C2 間, C2-C3 間, C1-C3 間) で相互に通信ができることを ping6 コマンドにて確認した．C1-C3 間の通信で C2 の L3 を通過していることを tcpdump にて確認した．

6.2 提案方式とエッジオーバレイ方式の TCP スループットの比較

図 12 に TCP スループットの測定環境を示す．アンダーレ

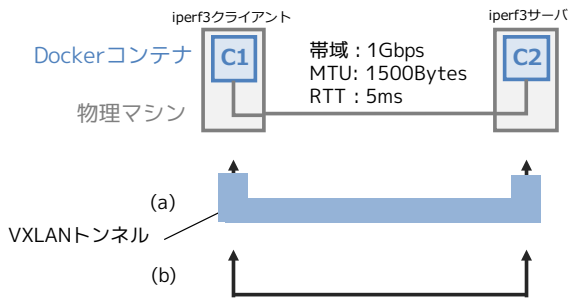


図 12 TCP スループットの測定環境

表 6 TCP スループットの測定結果

MSS	VXLAN (図 12 (a))	提案方式 (図 12 (b))
1440Bytes	334.3Mbps	489.0Mbps

インターネットの物理ノード (CPU : Intel Core i7-3770 3.40 GHz , メモリ : DDR3 1600MHz 8GB , OS : Ubuntu 16.04.3 LTS , カーネル : Linux Kernel 4.4.0-98-generic) 上に , ネットワークスライス上の仮想ノードである Docker コンテナ (OS : Ubuntu 16.04.3 LTS) を構築した . アンダーレイネットワークの物理ノード間のリンクは帯域 1G bps , MTU (Maximum Transmission Unit) 1500 バイト , RTT (Round-Trip Time) は 5ms である . 図 12 (a) は VXLAN によるネットワークスライス構築 , 図 12 (b) は本提案方式によるネットワークスライス構築の環境である .

2 台の仮想ノード間 (RAN A - UPF A 間の TCP スループットを , RAN A をクライアント , UPF A をサーバとして iperf3 により測定した . 2 つの環境でそれぞれフラグメンテーションが発生しない最大の MSS (Maximum Segment Size) の送信量でのスループットを測定した . 測定時間はそれぞれ 60 秒で , 各測定の送信開始から 3 秒間は計測データから除外した . それぞれ 10 回計測した平均値を表 6 に示す .

表 6 の測定結果から , カプセル化とカプセルの取り外しの処理はスループットの低下を引き起こすといえる .

7. おわりに

本稿では , 従来のネットワークスライス構築手法の問題を解決すべく , IPv6 アドレスの上位 8 バイトにネットワークスライスの識別子を埋め込むネットワークスライス構築手法を提案した . 提案方式によりネットワークスライスを構築し , トンネリングのないモビリティサポートプロトコルを採用することで , トンネリング処理を完全に撤廃した 5G ネットワークの構築が可能である . 概念実証の結果 , 本提案方式によるネットワークスライスが構築可能である見通しを得た . 今後は , UE のネットワークスライス選択手順の実装による概念実証 , ネットワークスライス作成機能などの手順の設計・実装をする必要がある .

文 献

[1] “M.2083 : IMT Vision - “Framework and overall objectives of the future development of IMT for 2020 and beyond””. <https://www.itu.int/rec/R-REC-M.2083-0-201509-I/>

[2] 落合孝壮 , 松枝耕平 , 金子晋文 , 寺岡文男 , “携帯通信網における Locator/ID 分離に基づくムービングセルの実現 ,” 情報処理

学会技術研究報告 , vol.2017-MBL-83 , no.20 , pp.1-8 , June 2017 .

[3] N. McKeown, T. Anderson, H. Balakrishnan, F. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, “OpenFlow: Enabling Innovation in Campus Networks,” ACM SIGCOMM Computer Communication Review, vol.38, no.2, pp.69-74, March 2008.

[4] M. Mahalingam, D. Dutt, K. Duda, P. Agarwal, L. Kreeger, T. Sridhar, M. Bursell, and C. Wright, “Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks,” RFC 7348, Aug. 2014.

[5] P. Garg and Y. Wang, “NVGRE: Network Virtualization Using Generic Routing Encapsulation,” RFC 7637, Sept. 2015.

[6] E. Rosen, A. Viswanathan, and R. Callon, “Multiprotocol Label Switching Architecture,” RFC 3031, Feb. 2001.

[7] B. Yi, B. Congxiao, and L. Xing, “FlowLAN: A non-tunneling distributed virtual network based on IPv6,” 2016 IEEE Information Technology, Networking, Electronic and Automation Control Conference, pp.229-234, May 2016.

[8] S. Ogawa, K. Yamazaki, R. Kawashima, and H. Matsuo, “T3: TCP-based High-Performance and Congestion-aware Tunneling Protocol for Cloud Networking,” 2016 IEEE Conference on Standards for Communications and Networking, pp.1-7, Oct. 2016.

[9] Z. Zhu and R. Wakikawa and L. Zhang, “A Survey of Mobility Support in the Internet,” RFC 6031, July 2011.

[10] S. Gundavelli and K. Leung and V. Devarapalli and K. Chowdhury and B. Patil, “Proxy Mobile IPv6,” RFC 5213, June 2008.

[11] J. Laganier, 樋口 健, 西田克利, “オール IP コアネットワークを実現する移動管理技術,” NTT DOCOMO テクニカル・ジャーナル, vol.17, no.3, pp.34-39, Oct. 2009 .

[12] W. Kiess, Y. X. Gu, S. Thakolsri, M. R. Sama, and S. Beker, “SimpleCore: a connectionless, best effort, no-mobility-supporting 5G core architecture,” 2016 IEEE International Conference on Communications Workshops, pp.367-372, May 2016.

[13] “3GPP”. <http://www.3gpp.org/>

[14] “System Architecture for the 5G System,” 3GPP TR 23.501 V1.3.0, Sept. 2017.

[15] “Study on control and user plane separation of EPC nodes,” 3GPP TR 23.714 V1.0.0, June 2016.

[16] “ドコモ 5G ホワイトペーパー,” Sept. 2014 .

[17] M. Ishiyama, M. Kunishi, K. Uehara, H. Esaki, and F. Teraoka, “LINA: A New Approach to Mobility Support in Wide Area Networks,” IEICE Transactions on Communications, vol.E84-B, no.8, pp.2076-2086, Aug. 2001.

[18] “Linux”. <https://www.linux.org/>

[19] “Docker”. <https://www.docker.com/>

[20] “LXC”. <https://linuxcontainers.org/>