Vol.2022-GN-115 No.42
Vol.2022-CDS-33 No.42
Vol.2022-DCC-30 No.42
2022/1/21

仮想IPv4アドレスを想定したCYPHONICアダプタの設計と基礎評価

後藤 廉[1,a)]　吉川 大貴[2,b)]　小村 聖[2,c)]　眞玉 和茂[1,d)]　内藤 克浩[1,e)]

概要：現在のインターネットには, Network Address Port Translation (NAPT) に伴う通信接続性の課題, IPv4 と IPv6 の非互換性による課題, 移動に伴うトランスポート層の通信切断の課題が存在する. また, ゼロトラストネットワークをはじめとした, 端末間のセキュア通信への着目が増加しており, 結果としてエンド間通信技術への需要も増加している. 上記の課題に関する個別の技術的解決策は多数報告されている一方, これらの問題を包括的に解決する手段については, 十分な技術的報告がなされていない. 著者らは, これらの諸問題を包括的に解決可能な技術としてオーバーレイネットワーク技術である CYber PHysical Overlay Network over Internet Communication (CYPHONIC) の開発を進めてきた. CYPHONIC では, 接続する端末 (CYPHONIC ノード) は, オーバーレイネットワーク上の通信を確立するための, 端末プログラムを導入する必要がある. しかし, 組込み型機器, 特定用途の専用サーバなど, 既存端末へ新たなプログラムを導入することが困難な端末が存在する. そこで, 本稿では既存機器に対して隣接設置するアダプタを開発することにより, オーバーレイネットワーク上の通信を実現する CYPHONIC アダプタを提案し, 概念実装を行う. CYPHONIC アダプタは, 一般の端末に対して CYPHONIC を介した通信に必要な処理を代行することにより, 既存の一般端末がオーバーレイネットワークへ参加することを可能とする. 概念実装では, 本アダプタの追加機能を既存の CYPHONIC ノード上に実装することにより, 提案技術が実現可能であることを確認した.

キーワード：通信接続性, 移動透過性, ゼロトラストネットワーク, オーバーレイネットワーク, IoT

1. はじめに

Internet of Things (IoT) は, センシング技術や無線通信とともに発展してきた. IoT デバイスは, センサーやアクチュエーション機能を備えた特定のサービスを実現するために, 様々な通信手段によって相互に接続される必要がある. IoT デバイスは, 一般に Internet Protocol (IP) を用いて相手端末と通信を行う. しかし, 爆発的に増加する IoT デバイスに反して, IPv4 アドレスの数は 2^{32} 個のみである [1], [2].

IP アドレスの枯渇問題に伴い, 現在のインターネットでは一般に Network Address Port Translation (NAPT) が用いられる. NAPT は, グローバル IP アドレスを多数のプライベート IP アドレスを用いて共有する仕組みを提供することにより, IPv4 グローバルアドレスの消費を軽減する [3]. 一方で, NAPT は外部のネットワークから NAPT 配下に存在する端末を隠蔽し, 外部ネットワークからの着信パケットを遮断する. そのため, 外部ネットワークに存在する端末は, NAPT 配下の端末に対して通信を開始することが極めて困難である [4]. この問題を NAPT 越え問題と呼び, NAPT 越えを実現するために, 様々な NAPT トラバーサル技術が提案されてきた [5], [6]. しかし, 既存の NAPT トラバーサル技術は通信に伴うシグナリングコストや処理遅延の増加, 経路冗長化などの課題が残されている [7].

また, IPv4 に続く次世代プロトコルとして, IPv6 の導入が進められている. IPv6 は, IPv4 に比べてアドレス空間が 2^{128} 個と拡張されているため, すべての端末に一意にアドレスを割り当てることが可能である [8]. しかし, IPv4 と IPv6 ではパケットの構造が異なるため, 互換性がなく, 相互に通信することが困難である [9]. さらに, 現在は IPv6

[1] 愛知工業大学情報科学部
Faculty of Information Science,
Aichi Institute of Technology, Nagoya, Aichi 464-0807, Japan
[2] 愛知工業大学大学院経営情報科学研究科
Graduate School of Business Administration and Computer Science,
Aichi Institute of Technology, Nagoya, Aichi 464-0807, Japan
[a)] r0719en@pluslab.org
[b)] yoshikawataiki@pluslab.org
[c)] hjr3ikmr@pluslab.org
[d)] matama@pluslab.org
[e)] naito@pluslab.org

への完全な移行に至っていないため, IPv4 と IPv6 が混在したネットワーク環境となっている. IPv4 と IPv6 の相互通信を実現する既存技術は, 双方の IP バージョンを共存させるため, 端末に改造が必要になることが課題となっている [10].

また, 近年の無線端末は複数の無線インターフェースを実装しており, ネットワークにアクセスする際にインターフェースを切替えて利用することが可能である [11]. しかし, IP では端末の移動について考慮されておらず, IP アドレスが保持する情報の二重性により, IP アドレスが変化するとトランスポート層の通信が切断される問題がある [12]. ネットワークが切り替えられた場合にも通信を継続可能な技術を移動透過技術と呼び, これまで様々な移動透過技術が提案されてきた. 既存の移動透過技術は, 端末が IPv6 をサポートしていることが前提としているものや, NAPT 配下の端末に対しては通信制限や冗長経路が発生するという問題がある [13], [14].

これらの端末間での相互通信では, 端末同士の接続性の確保に加え, セキュアな通信を実現することが求められる. そのため, 従来のネットワーク境界のセキュリティ対策だけでなく, 通信を行う端末を相互に認証し, 通信を行う際に送受信データの暗号化を行う必要がある [15], [16]. これらの課題に対して, 各課題への個別の解決策は多数提案されている一方で, 全課題を包括的に解決する枠組みについては, 議論が十分に行われていない. また, 具体的にこれらの課題を解決可能な概念実証についても十分な評価が行われていない.

著者らは, これらの課題を包括的に解決可能な技術としてオーバーレイネットワーク技術である CYber PHysical Overlay Network over Internet Communication (CYPHONIC) の提案を行うとともに, 開発を進めてきた [17], [18]. CYPHONIC は, 端末に仮想 IP アドレスを付与し, 実ネットワーク環境の影響を受けない, オーバーレイネットワーク上の通信をアプリケーションに対して提供する. 加えて, CYPHONIC は IPv4 と IPv6 通信間の変換をサポートしており, NAPT の有無に影響せず通信を確立し, さらに端末移動時に継続して通信が可能である. また, CYPHONIC 上で通信する端末は事前にクラウドサービスと通信することによって認証を行い, 通信時に暗号鍵を用いて送受信データを暗号化することで, セキュアなエンドツーエンド通信を実現可能である. 現状, CYPHONIC を介した通信を行うためには, 通信対象の端末に CYPHONIC の通信機能を持つプログラムを導入する必要がある. しかし, IoT デバイスや組込み型機器, 専用サーバなどの既存端末を想定した場合, 端末のリソース上の制限や, アプリケーション運用上の制約などから, 新たなプログラムの導入が困難なケースが見受けられる.

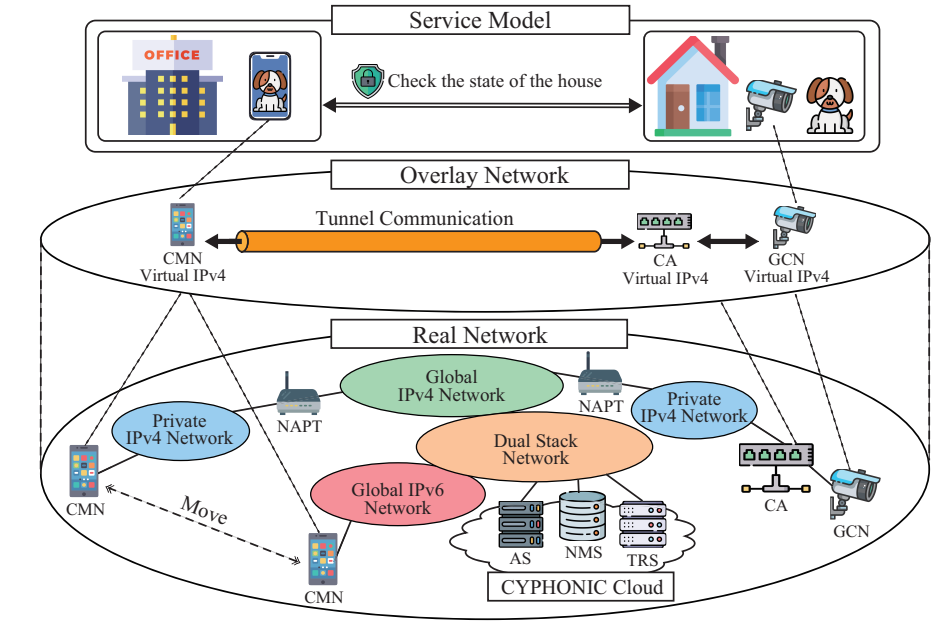
本研究では, これらの一般の既存端末（以下, 一般ノード と記述）に修正を加えることなく, 一般ノードがオーバーレイネットワークに参加可能とするためのアダプタ端末（以下, CYPHONIC アダプタ と記述）を提案する. CYPHONIC アダプタは, 一般ノードが CYPHONIC を介した通信に必要な処理を代行することにより, CYPHONIC の通信機能を持つプログラムを一般ノードに導入することなく, オーバーレイネットワークを用いた通信の実現が可能である. そのため, CYPHONIC アダプタを導入することにより, アタプタ配下の端末はセキュアなエンドツーエンド通信が可能である. 提案手法では, 多くの既存端末が IPv4 に対応していることを考慮し, 仮想 IPv4 アドレスを用いる. CYPHONIC アダプタを, 既存の CYPHONIC ノード上に実装することにより, 一般の端末に手を加えることなく, CYPHONIC の機能を付与できることを確認した. 本論文では, 一般ノードが CYPHONIC アダプタを用いて, CYPHONIC 上での通信を実現するためのシステム設計および概念実装について詳述する.

2. CYPHONIC の概要

図 1 に CYPHONIC の概要図を示す. CYPHONIC は, クラウドサービスと CYPHONIC を搭載した CYPHONIC ノードにより構成される. クラウドサービスには, Authentication Service (AS) , Node Management Service (NMS) , Tunnel Relay Service (TRS) の 3 種類のサービスが存在する. AS は, CYPHONIC ノードが正規のユーザであるかの認証処理を行う. また, AS は CYPHONIC ノードの識別子として Fully Qualified Domain Name (FQDN) を割り当て, NMS と CYPHONIC ノード間の通信を暗号化するための暗号鍵を配布する. NMS は, ネットワーク移動時にも不変な一意の仮想 IP アドレスを CYPHONIC ノードに割り当てる. また, NMS は CYPHONIC ノードの実 IP アドレスや NAPT の有無などのネットワーク情報を管理しており, 通信の際は双方のネットワーク環境に応じた適切な経路を選択し, 構築手順を指示する. TRS は, IPv4/IPv6 ネットワーク間の通信や NAPT を跨いだ通信などの直接通信が困難な際に, 両端末間の送受信データを中継する.

CYPHONIC ノードは, サービス起動時に AS に対して認証を行い, NMS に対してネットワーク位置情報の登録と, 仮想 IP アドレスの割り当てを受ける. CYPHONIC ノードは, 仮想 IP アドレスに基づいて通信を行うが, 実際の通信では実 IP アドレスを用いてすべてのパケットをカプセル化し, User Datagram Protocol (UDP) トンネルによる通信を行う. NMS が適切な通信経路を双方の CYPHONIC ノードに指示することにより, 通信経路上に NAPT が存在する場合や, IPv4/IPv6 デュアルスタック環境においても, 常に双方向の通信開始を実現可能である. また, CYPHONIC 上で通信を行う CYPHONIC ノードは, 互いに暗号鍵を直接交換することにより, 送受信データの暗号化を行うため,

Vol.2022-GN-115 No.42
Vol.2022-CDS-33 No.42
Vol.2022-DCC-30 No.42
2022/1/21



AS: Authentication Service　NMS: Node Management Service　TRS: Tunnel Relay Service
CA: CYPHONIC Adapter　CMN: CYPHONIC Mobile Node　GCN: General Correspondent Node

図 1　Overview of CYPHONIC

第三者に暗号鍵を知られることがないセキュアな通信が可能である．以下に構成要素の詳細を示す．

- Authentication Service
  AS は，CYPHONIC ノードの認証をアカウント情報または証明書を用いて行う．各 CYPHONIC ノードは，NMS との通信に暗号鍵を使用するため，AS は CYPHONIC ノードと NMS に暗号鍵を配布する．また，AS は CYPHONIC ノードが利用する一意な仮想 IP アドレスを管理していることから，NMS に仮想 IP アドレスの通知も行う．
- Node Management Service
  NMS は，CYPHONIC ノードに対し，トンネル通信に必要な仮想 IP アドレスを通知する．また，CYPHONIC ノードが NMS に送信するパケット情報を用いることにより，CYPHONIC ノードの実 IP アドレスや NAPT の有無から，接続されているネットワークの情報を管理する．通信の際は，双方から取得したネットワーク環境に応じて適切な経路を選択し，構築手順を指示する．
- Tunnel Relay Service
  TRS は，エンド間の直接通信が困難な場合に，中継処理を行うことで通信接続性を実現する．直接通信が困難な場合の具体例として，両 CYPHONIC ノードが NAPT 配下に接続される場合の通信，IPv4/IPv6 ネットワーク間での通信が挙げられる．
- CYPHONIC ノード
  CYPHONIC ノードは，相手ノードを FQDN によって識別し，NMS に所望の FQDN への通信開始を依頼する．また，NMS からの経路指示を受けることにより，所望のノードに対するトンネル通信を確立し，仮想 IP アドレスを用いた通信を行う．その際，両 CYPHONIC ノード間で暗号鍵を直接交換することにより，エンドツーエンドでのセキュアな通信を実現する．また，CYPHONIC ノードの移動に伴い物理 IP アドレスが変化した場合も，新たなトンネル通信を確立するとともに，仮想 IP アドレスを用いることで，継続的な通信を実現する．
- CYPHONIC アダプタ
  CYPHONIC アダプタは，既存端末に CYPHONIC ノードの機能を追加することが，さまざまな事情により困難な場合に利用するものである．CYPHONIC アダプタを既存機器に隣接設置することにより，CYPHONIC 上での通信に必要な処理を代行する．一般ノードは CYPHONIC アダプタに接続することで，オーバーレイネットワークへの参加が可能となる．また，CYPHONIC アダプタには，事前にクラウドサービスに登録された一般ノードのみが接続可能である．

3. 提案する CYPHONIC アダプタの概要

3.1 概要

本論文では，提案する CYPHONIC アダプタについて述べる．CYPHONIC は，CYPHONIC Daemon と呼ばれる端末プログラムをエンド端末に導入することでオーバーレイネットワーク上での通信を実現する．しかし，CYPHONIC Daemon を未導入な端末の場合は，CYPHONIC を用いたオーバーレイネットワーク上での通信を行うことが困難である．CYPHONIC を利用するエンド端末は，新たに

Vol.2022-GN-115 No.42
Vol.2022-CDS-33 No.42
Vol.2022-DCC-30 No.42
2022/1/21

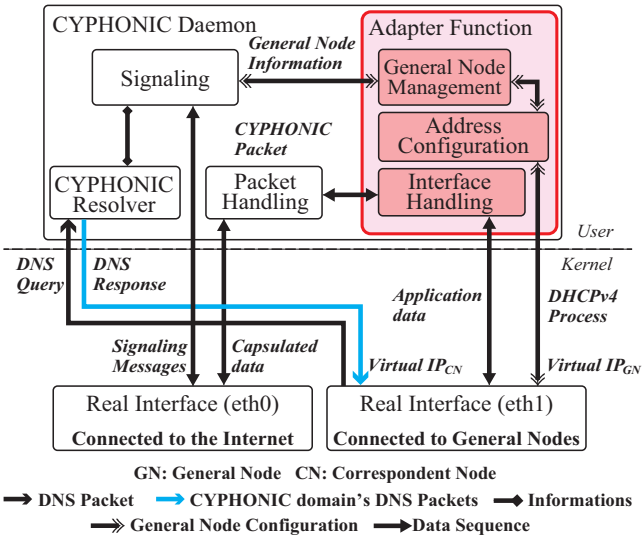


図 2 System model of CYPHONIC Adapter

CYPHONIC Daemon を導入することが困難な IoT デバイスや組込み機器, 専用サーバなどの既存端末を考慮しなければならない. そのため, これらの一般ノードをサポートするため, CYPHONIC の通信に必要な通信プロセスを代行するアダプタデバイスが必要である. CYPHONIC アダプタは, CYPHONIC Daemon に加え, 一般ノードに対する仮想 IP アドレスの付与, 一般ノードから届くパケットの操作, CYPHONIC での通信に必要な FQDN や暗号鍵の管理を行う. CYPHONIC アダプタは一般ノードに隣接設置することを想定しており, 一般ノードと CYPHONIC 上の相手ノードとの通信をブリッジする. CYPHONIC アダプタを実装する端末は, 物理 NIC を 2 枚用意し, 一方は, 一般ノードと接続し, 他方は, ネットワークと接続する. 一般ノードは隣接設置される CYPHONIC アダプタに接続することで, CYPHONIC 上での通信を確立することが可能である.

3.2 CYPHONIC Daemon の拡張

図 2 に, CYPHONIC アダプタのシステムモデル図を示す. CYPHONIC の機能は, ユーザ空間に存在する CYPHONIC Daemon により提供される. CYPHONIC Daemon は, ユーザ空間にバックグラウンドプロセスとして常駐し, 各サービスとシグナリングを行う機能, Domain Name System (DNS) パケットの処理を行う機能, 仮想 IP アドレスを用いたパケットのカプセル化およびデカプセル化処理を行う機能を実行する. CYPHONIC アダプタでは, CYPHONIC Daemon に加え, 一般ノードの管理機能, 一般ノードに対して仮想 IP アドレスの割り当てを行う機能, 一般ノードから届くパケットを CYPHONIC Daemon 内で処理するためのパケット処理機能, 一般ノードの FQDN や暗号鍵を管理する機能が必要である. これらの機能は, CYPHONIC Daemon 内に Adapter Function を追加することで実現する. 以下に, CYPHONIC Daemon が提供する機能, および Adapter Function の各モジュールが提供する機能について詳述する.

- CYPHONIC Daemon が提供する機能
  既存の CYPHONIC Daemon には, 以下のモジュールが存在し, CYPHONIC 上の通信に必要な処理を実行する.
  - Signaling Module
    端末が起動し, CYPHONIC Daemon が動作した際に, 最初に呼び出されるモジュールである. 前述した CYPHONIC のクラウドサービスである AS, NMS, TRS との一連のシグナリング処理を管理する.
  - Packet Handling Module
    定義された CYPHONIC のパケット形式に従って, パケットの生成, 暗号化および復号化, カプセル化およびデカプセル化処理を行う. また, 送信する CYPHONIC パケットの末端に改ざん検出を行うための Hash-based Message Authentication Code (HMAC) を付与する. 受信側は, 受信パケットの HMAC を計算することで, データの完全性を保証した通信を実現可能である.
  - CYPHONIC Resolver Module
    相手ノードのネットワーク情報をアプリケーションに伝達するために用いられる. 既存のアプリケーションが DNS を用いて通信を開始するのと同様に, CYPHONIC を用いたオーバーレイネットワーク上での通信においても FQDN を用いる. CYPHONIC Resolver Module は, 取得した相手ノードの仮想 IP アドレスから DNS パケットを生成する.
- Adapter Function の各モジュールが提供する機能
  Adapter Function は, 一般ノードをオーバーレイネットワークへ参加させるための諸機能を提供する.
  - General Node Management Module
    事前にクラウドサービスに登録されている一般ノードの情報を取得し, 管理するモジュールである. General Node Management Module は, 接続される一般ノードごとに仮想 IP アドレス, および FQDN と, 一般ノードを認識するための MAC アドレスを保持する. また, 一般ノードの通信プロセスごとに暗号鍵を生成し, 相手ノードと交換することで, アダプタ配下に存在する一般ノードを一意に識別したセキュアな通信を行うことが可能となる.
  - Address Configuration Module
    General Node Management Module から取得した情報に基づいて, 一般ノードへ仮想 IPv4 アドレスを付与する. CYPHONIC アダプタが付与するアドレスは, 一般ノードでは実 IP アドレスとして機能するが, CYPHONIC 上では一意な仮想 IP アドレスとして機

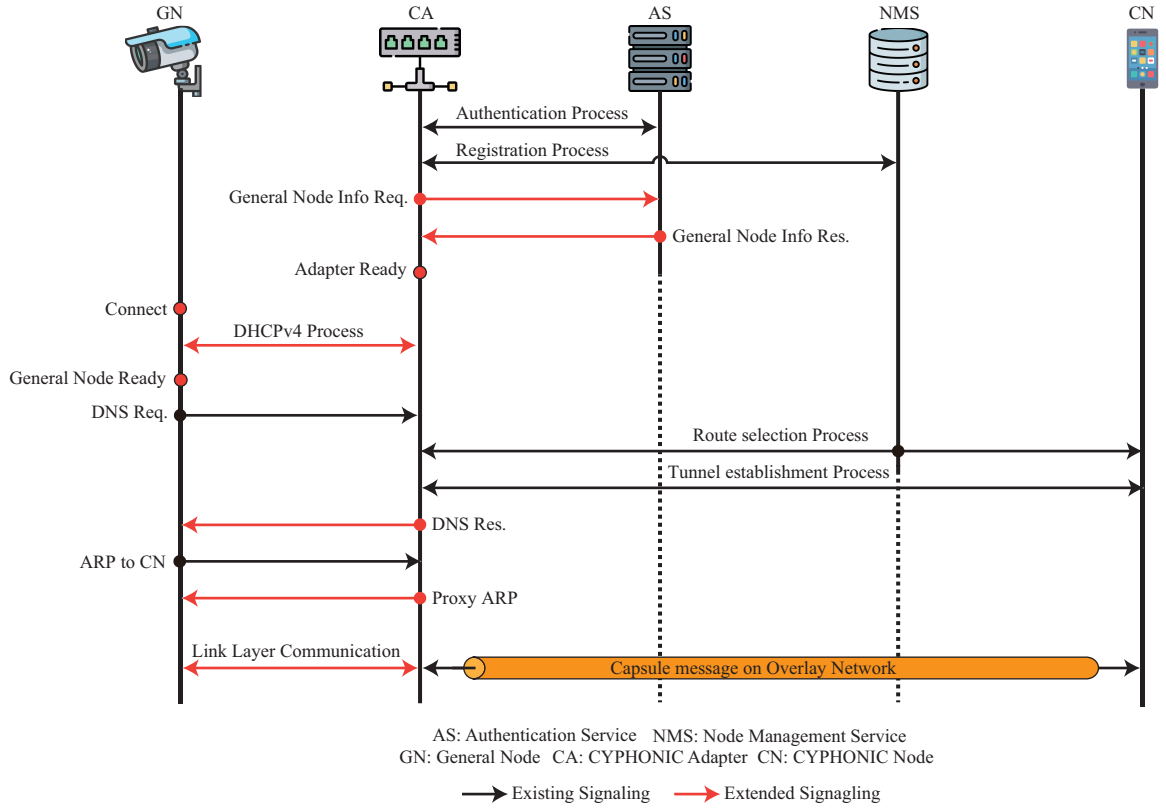


図 3 Communication sequence using CYPHONIC Adapter

能する．なお，クラウドサービスに登録されていない一般ノードが接続された場合は，仮想 IP アドレスの付与を行わない．この機能により，一般ノードは通常の Dynamic Host Configuration Protocol (DHCP) サービスを通して仮想 IPv4 アドレスの付与を受けることが可能となる．

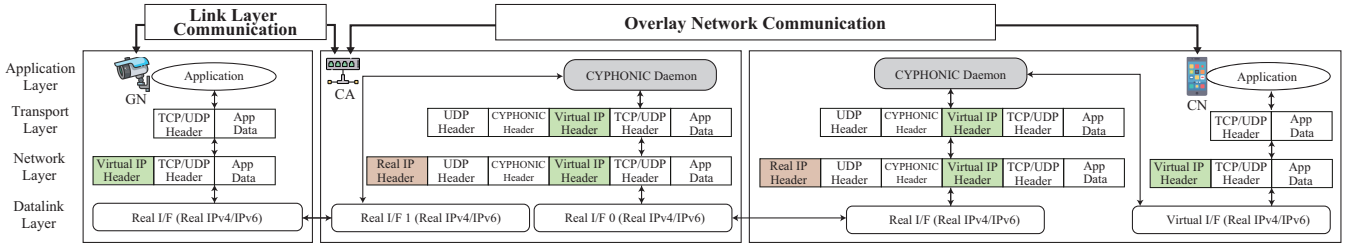
– Interface Handling Module
各クラウドサービスや相手ノードと通信をする際に，一般ノードが接続される実インターフェースを介して仮想 IP アドレス宛のパケットを取得する．取得したパケットは，Packet Handling Module へ受け渡すことにより処理する．また，Packet Handling Module からデカプセル化されたパケットを受け取った場合には，実インターフェースを介して一般ノードへ送信する．

3.3 通信シーケンス

一般ノードが CYPHONIC アダプタを用いて通信をする際のシーケンス図を図 3 に示す．CYPHONIC では，オーバーレイネットワークを用いたエンドツーエンド通信を行うために，認証処理，位置情報登録処理，経路選択処理，トンネル確立処理，データ通信処理の 5 つの処理が実行される．認証処理では，CYPHONIC 上のセキュア通信を実現するために，端末の認証を行う．位置情報登録処理は，端末が存在するネットワーク環境をクラウドサービスに通知し，仮想 IP アドレスを取得する．経路選択処理は，エンド端末から取得したネットワーク情報に基づいて，通信を行う際の経路選択を行う．トンネル確立処理は，オーバーレイネットワークの構築を行う処理であり，エンド端末間で通信に用いる暗号鍵の交換を行う．データ通信処理は，アプリケーションがオーバーレイネットワークを利用した通信を実現する際の処理である．

一般ノードは，事前にクラウドサービスに登録されていることを前提としている．従って，認証処理および位置情報登録処理は，CYPHONIC アダプタのみが実行する．また，CYPHONIC アダプタは接続される一般ノードの情報を取得するための，シグナリングを実行する．CYPHONIC アダプタがクラウドサービスから一般ノードの情報を取得すると，一般ノードを受け入れる準備が完了する．一般ノードは，CYPHONIC アダプタに接続することで，仮想 IP アドレスを取得すると，CYPHONIC 上での通信を行うことが可能となる．

一般ノードが，通信を開始する際，相手ノードの FQDN を指定して CYPHONIC アダプタに送信する．CYPHONIC アダプタは，一般ノードから DNS クエリを取得すると，経路選択処理を実行する．また，トンネル確立処理の際は，一般ノードを代行して暗号鍵を生成し，相手ノードと交換する．相手ノードの仮想 IP アドレスが一般ノードに通知された際，CYPHONIC アダプタが Proxy Address Resolution Protocol (ARP) を実行することにより，相手ノードへの



GN: General Node CA: CYPHONIC Adapter CN: CYPHONIC Node

図 4 Packet Flow

パケットを取得することが可能となる. 通信を実行する際, 一般ノードと CYPHONIC アダプタ間はリンクレイヤー通信を行い, CYPHONIC アダプタと相手ノードはオーバーレイネットワーク上でトンネル通信を行う. 以下に各処理について詳述する.

- 認証処理および位置情報登録処理
  CYPHONIC アダプタが起動すると, 自身の信頼性を確立するため, AS に対して認証処理を行う. また, CYPHONIC アダプタ自身の仮想 IP アドレスの取得と接続ネットワーク情報の登録をするため, NMS に対して位置情報登録処理を行う. CYPHONIC に接続する一般ノードは, 事前にクラウドサービスに登録済みであるため, これらの処理は CYPHONIC アダプタのみが実行する.
- 一般ノード情報取得処理
  CYPHONIC アダプタが管理する一般ノードの情報を取得するための処理である. CYPHONIC アダプタは, 予めクラウドサービスに登録されている一般ノードのみの接続を受け入れる. 従って, CYPHONIC はクラウドサービスから, これらの一般ノードの情報を取得する必要がある. 一般ノード情報取得処理により, 一般ノードが利用しているネットワークインターフェースの MAC アドレス, 割り当てる仮想 IP アドレス, 仮想 IP アドレスに対応する FQDN を管理可能となる.
- 仮想 IP アドレス割り当て処理
  一般ノードが CYPHONIC アダプタに接続された際, 仮想 IP アドレスの付与を行う処理である. 割り当てる仮想 IP アドレスは, 事前にクラウドサービスから取得した情報に基づいて行う. なお, 仮想 IP アドレスの割り当ては, 通常の DHCP と同様のプロセスによって行われる.
- 経路選択処理
  一般ノードが通信を開始する際, 相手ノードの FQDN を指定して CYPHONIC アダプタへ送信する. CYPHONIC アダプタが一般ノードから DNS クエリを受け取ると, 代理で経路選択処理を実行し, 相手ノードの仮想 IP アドレスおよび通信に用いる経路を取得する. また, 取得した仮想 IP アドレスから DNS パケットを生成して, 一般ノードへ送信する. CYPHONIC アダプタは, 一般ノードが相手ノードの MAC アドレスを要求した際に, Proxy ARP を実行し, 自身の MAC アドレスを応答する.
- トンネル確立処理
  オーバーレイネットワークの構築を行う処理であり, エンド端末間の通信を暗号化するための暗号鍵を交換する目的で行われる. 暗号鍵の生成は, 一般ノードを代行して CYPHONIC アダプタが実行する. また, 暗号鍵は通信プロセスごとに生成するため, 一般ノードからのパケットは CYPHONIC アダプタで暗号化され, セキュアな通信を実現可能である.
- データ通信処理
  一般ノードが CYPHONIC ノードと通信をする際のパケットフローを図 4 に示す. CYPHONIC アダプタは, 一般ノードが仮想 IP アドレス宛に送信したパケットを取得する. その後, 仮想 IP アドレスから生成したパケットを一般ノード用の暗号鍵を用いて暗号化し, CYPHONIC アダプタの実 IP アドレスでカプセル化して実ネットワークから送信する. 相手ノードに着信したパケットがデカプセル化されると, 一般ノードの仮想 IP アドレスが取り出されるため, アダプタ配下の端末を一意に識別した通信が実現可能である.

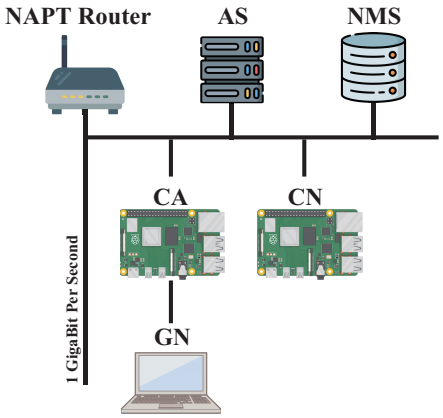
4. 実装

本研究では, 提案する CYPHONIC アダプタの概念実装を行う. 本実装では, 既存の CYPHONIC Daemon 内に一般ノードをサポートするための Adapter Function を追加する. 一般ノードと CYPHONIC アダプタ間の通信には, OS 標準のソケット API を用いる. CYPHONIC アダプタは仮想 IP アドレスの付与に, DHCP の機能を使用する. その際, デフォルトゲートウェイや DNS サーバの情報として CYPHONIC アダプタの仮想 IP アドレスを通知することにより, 一般ノードからのパケットを取得可能となる. 故に, CYPHONIC アダプタは一般ノードから送信されたパケットに対して, CYPHONIC パケットへのカプセル化処理や暗号化処理を行うことが可能となる.

また, 経路選択処理により, 一般ノードに対して相手ノー

表 1　Performance evaluation by UDP

	CYPHONIC Adapter			CYPHONIC Node		
Traffic	Throughput	Jitter	RTT	Throughput	Jitter	RTT
10Mbps	10Mbps	0.73ms	3.534ms	10Mbps	0.62ms	2.458ms
20Mbps	20Mbps	0.65ms	9.263ms	20Mbps	0.58ms	8.021ms
30Mbps	20Mbps	0.53ms	14.372ms	30Mbps	0.40ms	11.710ms



AS: Authentication Service　NMS: Node Management Service
GN: General Node　CA: CYPHONIC Adapter　CN: CYPHONIC Node

図 5　Evaluation model

表 2　Specifications of the measuring devices

CYPHONIC Cloud	
Machine	Raspberry Pi 4 Model B
OS	Raspbian GNU/Linux 10.0 (Buster)
CPU	Quad Core 1.5GHz Broadcom BCM2711 64bit
Memory	4GB RAM
CYPHONIC Node / CYPHONIC Adapter	
Machine	Raspberry Pi 4 Model B
OS	Raspbian GNU/Linux 10.0 (Buster)
CPU	Quad Core 1.5GHz Broadcom BCM2711 64bit
Memory	8GB RAM
General Node	
OS	macOS Big Sur Version 11.5
CPU	Dual Core 2.20GHz Intel(R) Core i7-5650U 64bit
Memory	8GB RAM

ドの仮想 IP アドレスが通知されることにより, CYPHONIC アダプタは MAC アドレスを解決するための ARP パケットを, 一般ノードから受信する. CYPHONIC アダプタは一般ノードから受信した ARP 要求に対して, 自身の MAC アドレスを代理で応答することにより, 一般ノードの ARP テーブルへマッピングさせる. この処理により, CYPHONIC アダプタは, 一般ノードが相手ノードへ送信するパケットを取得可能となるため, データ通信処理を代行することが可能となる. 以下に, Adapter Function が提供する各モジュールの実装について詳述する.

- General Node Management Module
  アダプタが管理する一般ノードの情報は, クラウドサービスから取得することを想定している. 一般ノードの情報として, 通信に用いる仮想 IP アドレス, および FQDN, 一般ノードを識別するための MAC アドレスを管理する. また, オーバーレイネットワーク上の通信を行う際には, 通信相手のノードとの間で利用する暗号鍵の管理も行う.
- Address Configuration Module
  一般ノードはオーバーレイネットワーク上で有効となる仮想 IP アドレスを利用する必要がある. そこで, CYPHONIC アダプタは DHCP の機能を活用することにより, 一般ノードの MAC アドレスを識別子として仮想 IP アドレスの割り当てを行う. また, 一般ノードの DNS クエリをトリガーにオーバーレイネットワーク上の通信を開始することから, デフォルトゲートウェイや DNS サーバの情報として, CYPHINC アダプタの仮想 IP アドレスの通知を行う.
- Interface Handling Module
  一般ノードとのパケット送受信を行う. 一般ノードが送信するパケットの宛先アドレスは通信相手の仮想 IP アドレスとなるため, Proxy ARP の機能を活用することにより, 通信相手ノードへのパケットを CYPHONIC アダプタが受信できるように処理を行う.

5. 検証および評価

本章では, 提案する CYPHONIC アダプタの動作検証および性能評価について述べる.

5.1 動作検証

提案方式の動作検証を行うにあたり, ping を用いた疎通確認を実施する. ping は, Internet Control Message Protocol (ICMP) における Echo Request パケットを対象ノードに送信し, 対象ノードから応答される Echo Reply を受信することで到達性を確認する. 表 1 に, 一般ノードから CYPHONIC ノードの FQDN 宛に ping を実行した際の RTT 値を示す. CYPHONIC アダプタが代理で経路選択処理およびトンネル確立処理, データ通信処理を実行することにより, 一般ノードに割り当てた仮想 IP アドレスを用いて, オーバーレイネットワーク上で ICMP を用いた疎通が可能であることを確認した. また, RTT 値を用いて, CYPHONIC ノード間の通信と比較を行った結果, 大きなオーバーヘッドが発生していないことを確認した.

5.2 性能評価

性能評価は図 5 に示すネットワーク構成を準備することにより実施した. また, 性能評価に利用したデバイスのスペック情報を表 2 に示す. 性能評価では, 提案する一般ノードと CYPHONIC ノード間の通信と, 比較対象として既存の CYPHONIC ノード間の特性を計測した. 測定では, スループット計測ソフトウェアである iperf3 と, RTT 測定が可能な ping を利用した.

表 1 に, UDP を利用してトラフィックを増加させた場合の特性を示す. 結果より, 30Mbps 程度のトラフィックを処理していることが確認される. また, RTT 及び Jitter も変動はしているが通信が十分に可能な範囲に収まっていることも確認できる. 本評価で利用したデバイスは暗号化処理をソフトウェア処理するものであり, AES-NI などの対応により, さらなる高速化も実現可能と考えている.

6. まとめ

本論文では, CYPHONIC を搭載することが困難な一般ノードへ対応するため, 既存端末に隣接設置する CYPHONIC アダプタを提案した. 概念実装では, 既存の CYPHONIC Daemon に一般ノードをサポートするための機能を追加することで, CYPHONIC アダプタを実現した. 提案手法により, CYPHONIC アダプタが CYPHONIC を介した通信に必要な処理を代行することで, 一般ノードに手を加えることなく, CYPHONIC の機能を付与可能であることを確認した. また, CYPHONIC アダプタの検証評価より, 一般ノードが CYPHONIC アダプタを用いてオーバーレイネットワーク上での通信を確立可能であることを確認した.

謝辞 本研究の一部は JSPS 科研費 (21K11877) の助成を受けたものである. 記して謝意を表する.

参考文献

[1] Guo, H. and Heidemann, J.: Detecting IoT Devices in the Internet, *IEEE/ACM Trans. Netw.*, Vol. 28, No. 5, p. 2323–2336 (online), DOI: 10.1109/TNET.2020.3009425 (2020).

[2] Boulevard, W.: Internet protocol, *The Internet Engineering Task Force (IETF) RFC 791* (1981).

[3] Holdrege, M. and Srisuresh, P.: IP Network Address Translator (NAT) Terminology and Considerations, RFC 2663 (1999).

[4] Huang, C.-L. and Hwang, S.-H.: The asymmetric NAT and its traversal method, *2009 IFIP International Conference on Wireless and Optical Communications Networks*, pp. 1–4 (online), DOI: 10.1109/WOCN.2009.5010536 (2009).

[5] Yang, L. and Lei, K.: Combining ICE and SIP protocol for NAT traversal in new classification standard, *2016 5th International Conference on Computer Science and Network Technology (ICCSNT)*, pp. 576–580 (online), DOI: 10.1109/ICCSNT.2016.8070224 (2016).

[6] Huang, F., Yu, L., Shen, T. and Hu, S.: The P2P Solution Research and Design Based on NAT Traversing Technology, *2019 IEEE 3rd Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)*, pp. 1347–1351 (online), DOI: 10.1109/IMCEC46724.2019.8984136 (2019).

[7] Mikamo, Y., Asahi, K., Suzuki, H. and Watanabe, A.: Proposal for an ad-hoc routing protocol considering traffic conditions and evaluation of UDP using a redundant route, *2014 Seventh International Conference on Mobile Computing and Ubiquitous Networking (ICMU)*, pp. 72–73 (online), DOI: 10.1109/ICMU.2014.6799062 (2014).

[8] Hinden, B. and Deering, D. S. E.: Internet Protocol, Version 6 (IPv6) Specification, RFC 2460 (1998).

[9] Wu, P., Cui, Y., Wu, J., Liu, J. and Metz, C.: Transition from IPv4 to IPv6: A State-of-the-Art Survey, *IEEE Communications Surveys Tutorials*, Vol. 15, No. 3, pp. 1407–1424 (online), DOI: 10.1109/SURV.2012.110112.00200 (2013).

[10] Durand, A., Droms, R., Lee, Y. and james woodyatt: Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion, RFC 6333 (2011).

[11] Fafolahan, E. M. O. and Pierre, S.: A Seamless Mobility Management Protocol in 5G Locator Identificator Split Dense Small Cells, *IEEE Transactions on Mobile Computing*, Vol. 19, No. 8, pp. 1745–1759 (online), DOI: 10.1109/TMC.2019.2915071 (2020).

[12] Park, J.-T., Chun, S.-M., Choi, J.-H. and Lee, S.-M.: Simple mobility management protocol for global seamless handover, *2012 IEEE Consumer Communications and Networking Conference (CCNC)*, pp. 677–681 (online), DOI: 10.1109/CCNC.2012.6181031 (2012).

[13] Giaretta, G.: Interactions between Proxy Mobile IPv6 (PMIPv6) and Mobile IPv6 (MIPv6): Scenarios and Related Issues, RFC 6612 (2012).

[14] Tong, H., Wang, T., Zhu, Y., Liu, X., Wang, S. and Yin, C.: Mobility-Aware Seamless Handover With MPTCP in Software-Defined HetNets, *IEEE Transactions on Network and Service Management*, Vol. 18, No. 1, pp. 498–510 (online), DOI: 10.1109/TNSM.2021.3050627 (2021).

[15] Sicari, S., Rizzardi, A., Grieco, L. and Coen-Porisini, A.: Security, privacy and trust in Internet of Things: The road ahead, *Computer Networks*, Vol. 76, pp. 146–164 (online), DOI: https://doi.org/10.1016/j.comnet.2014.11.008 (2015).

[16] Yao, Q., Wang, Q., Zhang, X. and Fei, J.: Dynamic Access Control and Authorization System Based on Zero-Trust Architecture, *2020 International Conference on Control, Robotics and Intelligent System*, CCRIS 2020, New York, NY, USA, Association for Computing Machinery, p. 123–127 (online), DOI: 10.1145/3437802.3437824 (2020).

[17] Yoshikawa, T., Isomura, S., Komura, H., Kubota, S., Nishiwaki, C. and Naito, K.: Performance evaluation of shared library supporting multi-platform for overlay network protocol, *2020 IEEE 9th Global Conference on Consumer Electronics (GCCE)*, pp. 777–781 (online), DOI: 10.1109/GCCE50665.2020.9292015 (2020).

[18] Yoshikawa, T., Komura, H., Nishiwaki, C., Goto, R., Matama, K. and Naito, K.: Evaluation of New CYPHONIC: Overlay Network Protocol Based on Go Language, *2022 IEEE International Conference on Consumer Electronics (ICCE) (2022 ICCE)*, Las Vegas, USA, pp. 1–6 (2022).