情報処理学会論文誌 Vol. 51 No. 9 1234–1241 (Sep. 2010)

端末の改造が不要なNAT越え通信システムNTSSの提案と評価

宮崎 悠[†1,*1] 鈴木秀和[†1,†2,*2] 渡邊 晃[†1]

インターネット側の端末からプライベートアドレスの端末に通信を開始できないというNAT越え問題は，IPv4の汎用性を損なう大きな要因となっている．これまでに研究されたNAT越え問題の解決技術は，多くの場合ユーザ端末に機能を実装する必要があった．しかし，端末の多様化に伴い，端末側に機能追加が難しい場合も考えられる．そこで本論文では，DNSサーバとNATルータを改造し，両者が連携することにより，ユーザ端末に機能を実装することなくNAT越えを実現する方式を提案する．提案方式は，外部ノードがNAT配下のノードに通信を開始する際，DNSサーバがNATルータに指示を与えることにより，NATテーブルの生成を行う．プロトタイプシステムの実装を行い，性能評価を行った結果，通信開始時のオーバヘッドはほとんど発生せず，通常のNATルータと同等のスループットを実現できることを確認した．

Proposal and Evaluation of NAT Traversal Support System Independent of User Terminals

Yutaka Miyazaki,[†1,*1] Hidekazu Suzuki[†1,†2,*2] and Akira Watanabe[†1]

NAT traversal problem, that is, a terminal on the Internet can not start communication against a terminal in the private address network, is the major reason for spoiling the versatile of IPv4 networks. Technologies those tried to solve the problem, in many cases, have to remodel end terminals. However, accompanied with diversification of terminals, it might be difficult to remodel the terminal in some cases. In this paper, NTSS (NAT Traversal Support System), in which a DNS server and a NAT router cooperate each other, and make the NAT create a proper NAT table, without remodeling end terminals, is proposed. We have implemented the NTSS and confirmed the good performance.

1. はじめに

IPv4ネットワークではIPアドレスの枯渇を回避するため，家庭内や企業内のネットワークはプライベートアドレスで構築するのが一般である．プライベートアドレスのネットワークとインターネットの間には必ずアドレス変換装置（以下NAT：Network Address Translator）が必要である．しかし，このような環境では，インターネット側の端末からプライベートアドレス側の端末へ通信を開始できないという制約があり，NAT越え問題と呼ばれている．これまでのインターネットの利用形態はWWWの閲覧やメールの利用など，一般にグローバルアドレス空間に設置されたサーバに対してプライベートアドレス空間に存在する端末側から通信を開始していた．ファイアウォールでもこのような通信形態のみを許可するのが一般的であったため，NATの制約が表面化することはなかった．しかし，近年ではネットワークインフラの普及に伴い，家庭内にもネットワークを構築する例が増加しており，外出先から家庭内のサーバなどに自由にアクセスしたいというニーズが十分に考えられる．

IPv4アドレスの枯渇を根本的に解決するための手段としてIPv6が検討されてきたが，IPv4が既に広く浸透しており，IPv6技術の導入は当初想定していたように進んでいない．また，IPv6の導入が始まったとしてもIPv4/IPv6の混在環境が当分続くことが想定され，NATの利用は今後も避けられない．今後の利用形態の多様化を考慮すれば，IPv4におけるNATの制約を除去することは有益である．

NATのアドレス変換テーブル（NATテーブル）は，原理的にプライベートアドレス空間からグローバルアドレス空間へのアクセス開始時にのみ生成される．また，グローバルアドレス空間からはNAT装置のIPアドレスしか見えないため，NATの内側のノードを指定することができない．このような制約を除去するために，NATテーブルを予め設定しておく静的NATがあり，多くの製品で設定方法が提供されているが[1)]，ポート番号1つに対して1台の内部ノードしか設定できないうえ，動的に変更できないため汎用性に欠ける．な

†1 名城大学大学院理工学研究科
Graduate School of Science and Technology, Meijo University
†2 日本学術振興会特別研究員 PD
Research Fellow of the Japan Society for the Promotion of Science
*1 現在，KDDI株式会社
Presently with KDDI Corporation
*2 現在，名城大学理工学部
Presently with Faculty of Science and Technology, Meijo University

お，企業や家庭などのプライベートネットワークにリモートアクセスする方法として VPN もあるが，NAT 越え技術ではないため本論文の範囲外とする．

NAT 越え問題を汎用的に解決する為にこれまで様々な解決手法が提案されてきたが，その目的により以下のように分類することができる．すなわち，既存の NAT 装置をそのまま使えることを目的としたアプリケーションレベル改造方式，既存のアプリケーションをそのまま使えることを目的としたネットワークレイヤ改造方式，端末の改造を不要とすることを目的とした端末非依存方式である．

アプリケーションレベル改造方式は，エンド端末のアプリケーションとインターネット上に設置したサーバが NAT テーブルの情報を交換し，NAT に生成された NAT テーブルに合わせて，外部端末からパケットを送信する点が特徴である．この方式は，アプリケーションが限定されることと，第三の装置が必要になるという課題がある．代表例として，STUN[2),3)]，TURN[4)]，UPnP[5)] などがある．

ネットワークレイヤ改造方式は，アプリケーションを限定しないために，外部端末のカーネルや NAT ルータなどのネットワーク機器に手を加える．外部端末と NAT ルータが協調してパケットを内部に転送する点が特徴である．この方式は，端末のカーネルを改造するため，OS ごとに異なる対応が必要となる．代表例として，4+4[6)]，NAT-f[7)] などがある．

端末非依存方式は，DNS サーバ，NAT ルータ，あるいはインターネット上のサーバなどが情報交換し，一般端末が送信するパケットを通信経路上でアドレス変換し，プライベートアドレス空間の中に転送する点が特徴である．この方式は研究事例がそれほど多くないため，研究途上であるといえる．端末非依存方式の AVES[8)] では，第三の装置が必要であること，通信経路が冗長になること，送信元アドレスが実際と異なるため経路上のルータで廃棄される可能性があるなどの課題がある．

情報家電やモバイル端末の多様化により，今後はユーザが自由に端末に機能を追加できない場合も考えられる．また，機能追加が可能でも一般のユーザにとっては導入が困難な場合が多い．そこで，ここでは端末に改造が不要な端末非依存方式に着目する．本論文では，一般端末の通信開始時における名前解決の際に DNS サーバと NAT ルータが情報交換し，DNS サーバから NAT ルータに対して適切な NAT テーブルの生成を指示することにより NAT 越えを可能とする NTSS（NAT Traversal Support System）を提案する．この方式は，一般端末の通信パケットに合わせて NAT テーブルをオンデマンドに生成する点に特徴があり，第三の装置の助けを借りることなくエンドエンドの通信が可能である．なお，本論文で言う第三の装置とは，システムを実現するために新たに導入した装置を指し，実環境で既に稼働済みの DNS サーバや NAT などに改造を加えて実現できるものは含めない．

提案方式を FreeBSD 上に実装し，動作検証および性能測定を行った．DNS サーバによる名前解決時のオーバヘッドおよびエンドノード間の通信のスループットを評価した結果，事実上問題ない性能を有することを確認した．

以降，2 章で提案技術と目的が同じ AVES について詳細に説明し，その課題を述べる．3 章で提案技術を説明し，4 章で実装について述べる．5 章で提案方式の動作検証と性能評価の結果を示し，最後に 6 章でまとめる．

2. 既存の端末非依存方式の課題

既存の端末非依存方式の例として AVES の詳細と課題について述べる．以降，NAT の外側（グローバルアドレス空間側）に存在するノードを EN（External Node），NAT の内側（プライベートアドレス空間側）に存在するノードを IN（Internal Node）と表記する．

図 1 に AVES の動作を示す．AVES ではインターネット上に AVES 対応 DNS サーバ（以下 ADNS サーバ）と Waypoint と呼ばれる専用の中継装置を配置する．また，IN 側ローカルネットワークを構築する NAT ルータも AVES に対応する必要がある（以下 ANAT ルータ）．IN は ADNS サーバに自身の名前（*alice*）とプライベート IP アドレス（*PA*1）に加え，ANAT ルータのグローバル IP アドレス（*GA*2）を関連づけて登録しておく．EN は改造が不要であるが，プライマリ DNS として ADNS サーバを設定しておく必要がある．

EN が ADNS サーバに IN（*alice*）の IP アドレスを問い合わせると，ADNS サーバは

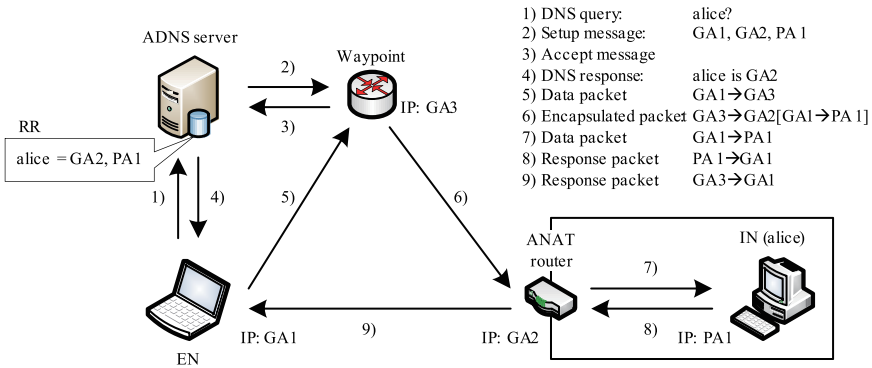


図 1　AVES の動作
Fig. 1　Behavior of AVES.

Waypoint に Setup メッセージを送信する．Setup メッセージには EN のグローバル IP アドレス（$GA1$），ANAT ルータのグローバル IP アドレス（$GA2$）および *alice* のプライベート IP アドレス（$PA1$）が含まれる．Waypoint はこれを受理したら，経路変換テーブルを生成して ADNS サーバに Accept メッセージを応答する．ADNS サーバは Waypoint のグローバル IP アドレス（$GA3$）を EN に応答する．このため，EN は *alice* 宛の通信パケットを Waypoint に対して送信することになる．

Waypoint は EN からのパケットを受信すると，経路変換テーブルに基づいて宛先アドレスを *alice* のプライベート IP アドレス（$PA1$）に変換する．さらに ANAT ルータ宛の IP ヘッダで変換したパケットをカプセル化して，ANAT ルータへ送信する．ANAT ルータは上記パケットを受信すると，カプセル化を解除し *alice* へ転送する．これに対し，*alice* からの応答パケットは Waypoint を経由せず，ANAT ルータから EN へ直接送信される．ここで，ANAT ルータは送信元 IP アドレスを Waypoint の IP アドレスとなるように変換する．このようにして EN から IN の通信は Waypoint を経由し，IN から EN への応答は直接通信という三角経路となる．

AVES は Waypoint という第三の特殊な装置が必要であるため，この装置が故障した場合の対策を別途考える必要がある．また，経路が冗長になることや，カプセル化によるパケット冗長が発生し，スループットが低下するなどの課題がある．更に，IN からの応答パケットの送信元アドレスが ANAT ルータではなく Waypoint の IP アドレスとなるため，ネットワーク上のルータにイングレスフィルタ[⋆1]などのセキュリティが設定されている場合，パケットが経路途中で破棄される可能性がある．

3. 提案方式

提案方式は，エンドツーエンド通信の利点を損なうことなく NAT 越え通信を実現することができる．提案方式を NTSS（NAT Traversal Support System），実行するプロトコルを NTS プロトコルと呼ぶ．また DNS サーバと NAT ルータを改造し，NTS プロトコルを実装する．改造した DNS サーバを NTS サーバ，改造した NAT ルータを NTS ルータと呼ぶ．NTS ルータは NTS サーバと協調し，外部から送信されてくるパケットに合わせて NAT テーブルをオンデマンドで生成する点が特徴である．

今回のシステムにおいては，汎用性を向上させるため，EN と IN の DNS サーバが異なっていてもよいことを想定した．DNS の役割として，端末の位置を登録するアドレス登録機能と，通信開始装置からのクエリを受けて相手装置のアドレスを取得するアドレス解決機能の 2 つがある．両者の機能は AVES のように一つの DNS サーバで実現することも可能であるが，以下の説明においては EN のアドレス解決用 DNS サーバと IN のアドレス登録用 DNS サーバが別である一般的な DNS システムを想定して記述した．

3.1 ネットワーク構成と事前設定

NTSS を実現する構成機器を**図 2** に示す．インターネット上に EN のアドレス解決用 DNS サーバとなる NTS サーバと，IN のアドレス登録用 DNS サーバとなる Dynamic DNS（以下 DDNS）サーバを設置する．DDNS サーバは既存のサービスプロバイダが提供している実際のサーバを利用することができ，NTSS を実現するための機能は不要である．ここで EN，NTS ルータのグローバル IP アドレスをそれぞれ $GA1$，$GA2$ とし，IN（*alice* と *bob*）のプライベート IP アドレスをそれぞれ $PA1$，$PA2$ とする．

事前設定として，DDNS サーバには IN の名前と NTS ルータのグローバル IP アドレスの対応関係を登録する．NTS ルータには IN の名前とプライベート IP アドレスの対応関係を，PHL（Private Host List）と呼ぶテーブルに登録する．また，EN のプライマリ DNS として NTS サーバを設定しておく．

以下に EN から IN（*alice*）へ通信を開始する場合を例に挙げ，名前解決時と通信開始時に分けて説明する．

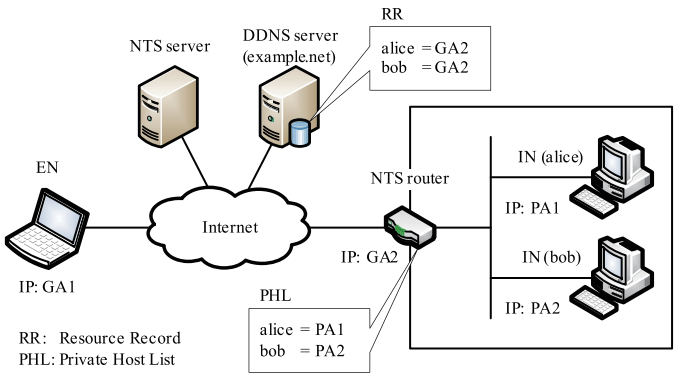


図 2　NTSS を実現する構成機器
Fig. 2　Deveices that realize NTSS.

⋆1 ルータが IP パケットを転送する際に，パケットの送信元 IP アドレスを見て転送許可の判断を行う仕組み

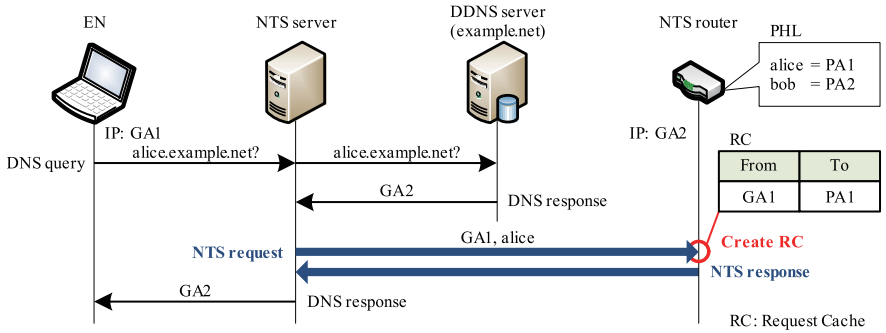


図 3　名前解決シーケンス
Fig. 3　Name resolution sequence.

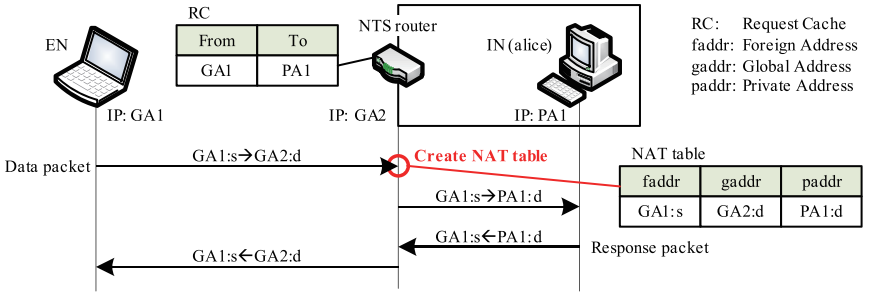


図 4　通信開始シーケンス
Fig. 4　Initial communication sequence.

3.2　名 前 解 決

図 3 に名前解決シーケンスを示す．EN は通信を開始するに当たり，*alice* の名前解決を NTS サーバへ依頼する．NTS サーバは通常の DNS の仕組み[⋆1]により，*alice* を管理する DDNS サーバより NTS ルータの IP アドレス（$GA2$）を取得する．NTS サーバはこの名前解決結果を EN へ応答する前に，NTS 要求メッセージを NTS ルータに送信する．NTS 要求メッセージの目的は，IP アドレス “$GA1$” が割り当てられている EN から *alice* への接続要求があることを NTS ルータに通知することである．この通知を受け取った NTS ルータは PHL を参照し，*alice* のプライベート IP アドレス（$PA1$）を取得する．その後，EN と IN の IP アドレスの関係を RC（Request Cache）へ記憶して，NTS サーバへ NTS 応答メッセージを返信する．応答メッセージを受信した NTS サーバは，先ほど取得した名前解決結果（$GA2$）を EN に応答する．

3.3　通 信 開 始

図 4 に，名前解決後の通信開始シーケンスを示す．EN は名前解決の結果，*alice* の IP アドレスを “$GA2$” と認識しているため，NTS ルータに向けて通信を開始する．ここで，

$$GA1 : s \rightarrow GA2 : d \tag{1}$$

は送信元 IP アドレス $GA1$，送信元ポート番号 s，宛先 IP アドレス $GA2$，宛先ポート番号 d の通信であることを示す．s は EN のカーネルが選択した任意のポート番号，d は IN がサービスを提供しているポート番号である．

NTS ルータはインターネット側からパケットを受け取ると，送信元 IP アドレスをキーとして RC の内容を確認する．RC に該当するデータがあれば，NTS ルータは受信したパケットの内容と RC の内容から次のような NAT テーブルを動的に生成する．

$$GA1 : s \leftrightarrow \{GA2 : d \Leftrightarrow PA1 : d\} \tag{2}$$

上記 NAT テーブルの意味は，外側トランスポートアドレス[⋆2] “$GA1 : s$” との通信では NAT のトランスポートアドレス “$GA2 : d$” と IN のトランスポートアドレス “$PA1 : d$” が対応していることを意味する．即ち，“$GA1 : s$” から “$GA2 : d$” へ送信されたパケットは，NTS ルータの NAT 機能において宛先が “$PA1 : d$” に変換されて *alice* へ転送される．これに対する *alice* からの応答パケットは上記と逆の変換を行い，EN へ送信される．

以上の手順により，EN から IN に対して NAT 越え通信を開始することができる．

3.4　提案方式の利点

NTSS は既存の端末非依存方式（AVES）で必要であった，EN から IN への送信パケットを転送するための第三の装置が不要であり，図 4 以降の通信フェーズにおいては，EN と IN 間でエンドエンドの通信を行うことができる．そのため，AVES の Waypoint に必要であった二重化や負荷分散の仕組みなどを検討する必要がない．インターネット上で転送される通信パケットのアドレスはエンドエンドのアドレスを正しく示しており，AVES のよう

⋆1 NTS サーバは反復クエリを使用して，ルートネームサーバから順に名前の解決を行うが，図 3 ではその部分を省略している．

⋆2 NTS ルータから見た外部からの通信開始側端末の IP アドレスとポート番号の組．

にパケットが途中のルータで破棄される心配はない．このように，NTSS は既存の端末非依存型 NAT 越えシステムの課題を解決している．

IN が複数存在し，同じポート番号でサービスを提供（例えば *bob* も *d* 番ポートでサービスを提供）していた場合は，NTS ルータ側のトランスポートアドレス “*GA*2 : *d*” が一致してしまう．しかし，EN 側の外部トランスポートアドレスのポート番号が異なるため，NTS ルータは *alice* 宛の通信と *bob* 宛の通信を区別することができ，問題なく通信が可能である．

4. 実装方法

NTSS の検証を行うため，プロトタイプシステムとして NTS サーバ機能を FreeBSD 上のアプリケーションとして，NTS ルータ機能を FreeBSD の NAT デーモン内に実装した．

4.1 NTS サーバの実装

図 5 に NTS サーバの実装概要を示す．図中の BIND は DNS アプリケーションのプログラムであり，NTS サーバモジュールは今回実装したデーモンである．NTS サーバでは BIND を通常の 53 番ポートとは異なる任意のポート（ここでは 10,053 番）でリッスンするように変更し，NTS サーバモジュールが DNS パケットを BIND に変わって送受信するために，53 番ポートでリッスンする．図中において “Ephemeral” と記載されているポートは，パケット送信時に OS により決定される任意のポート番号であり，FreeBSD では 49,152〜65,535 の範囲から割り当てられる．

NTS サーバモジュールは DNS メッセージを受信すると，BIND へそのまま渡す．BIND は通常の DNS 機能により名前解決を行い，その応答結果を NTS サーバモジュールへ返す．次に，NTS サーバモジュールは NTS ルータに対して NTS 要求メッセージを送信し，応答メッセージを受信した後に DNS 応答結果を EN に返信する．NTS サーバモジュールは DNS メッセージを一意に識別するためのトランザクション ID により，NTS ルータ間で行うネゴシエーションを識別する．上記手順により，NTS サーバは EN から見ると通常の DNS サーバに見える．

4.2 NTS ルータの実装

図 6 に NTS ルータの実装概要を示す．natd は FreeBSD のユーザランドで動作する NAT デーモンであり，NTS ルータモジュールはその内部に実装される．ipfw はカーネルで動作するファイアウォールのモジュールであり，Divert ソケットを利用して natd との間でパケットの受け渡しを行う．Divert ソケットとは，IP パケットの入出力を迂回させて，ユーザランドまで持って行くことが出来る機構である．この方法により natd が持つ NAT としての様々な機能をそのまま利用することができる．

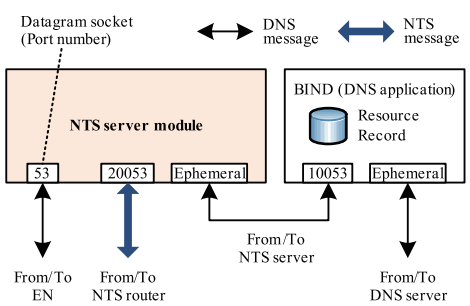


図 5　NTS サーバの実装概要
Fig. 5　Implementation of NTS server.

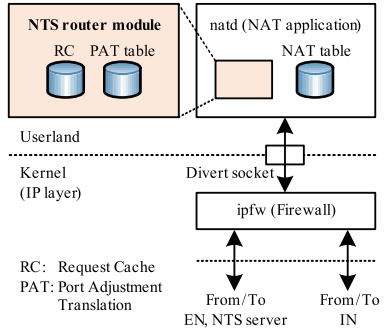


図 6　NTS ルータの実装概要
Fig. 6　Implementation of NTS router.

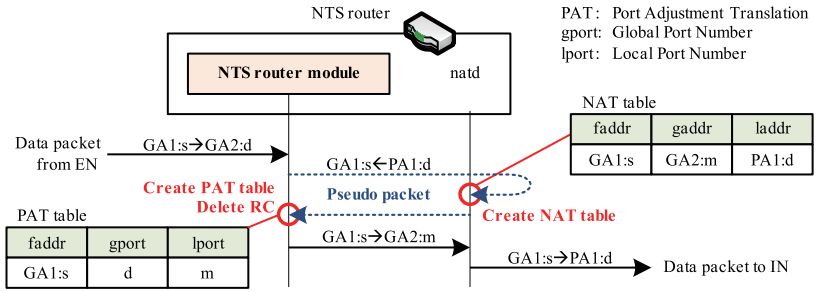


図 7　NAT テーブルの生成方法と PAT テーブルとの連携変換
Fig. 7　NAT table generating technique and translation in cooperation with PAT table.

natd は Divert ソケットからパケットを受信し，NTS ルータモジュールへ渡す．NTS ルータモジュールはパケットの種類を確認し，NTS サーバとのネゴシエーションや受信パケットの変換処理等の機能を実現する．NTS 要求メッセージであれば NAT テーブルおよび PAT（Port Adjustment Translation）テーブルを生成する．PAT テーブルについては，4.3 節で詳述する．その他のパケットであれば PAT テーブルと NAT テーブルに基づいたアドレス変換処理が行われる．

4.3 NAT テーブルの生成方法

NTS ルータの NAT テーブルを生成するにあたり，以下のような工夫をした．**図 7** に NAT

テーブルの生成方法を示す．NTS ルータは EN からの通信パケット “$GA1 : s \rightarrow GA2 : d$” を受信すると，NTS ルータモジュールで PAT テーブルを検索する．初めて受信した場合は PAT テーブルがまだ存在していないので，名前解決時に生成されている RC を参照する．ここで，RC に記憶されている IN のプライベート IP アドレス（$PA1$）と受信パケットの送信元トランスポートアドレス（$GA1 : s$）から，疑似パケットと呼ぶ擬似的なパケットデータを作成する．疑似パケットは送信元が “$PA1 : d$”，宛先が “$GA1 : s$” のような形式となっており，これを natd の処理に渡す．これにより，natd はあたかも IN から EN 宛のパケットを受信したものと認識して，以下のような NAT テーブルを生成する．

$$GA1 : s \leftrightarrow \{GA2 : m \Leftrightarrow PA1 : d\} \tag{3}$$

ここで m は通常 NAT が動的にマッピングする任意のポート番号である．

ただし，EN からのパケットの宛先は “$GA2 : d$” であるため，式 (3) の NAT テーブルのままではポート番号が一致しない．そこで NTS ルータモジュールにおいて，次のような PAT テーブルを生成して NAT テーブルの不整合性を解消する．

$$GA1 : s \leftrightarrow \{* : d \Leftrightarrow * : m\} \tag{4}$$

上記 PAT テーブルの意味は，送信元が “$GA1 : s$” で宛先ポート番号が d 番のパケットについて，宛先ポートを m に変換することを示している．すなわち，式 (4) の変換後に式 (3) の変換を行うことにより，式 (2) の変換と等価になる．IN からの応答パケットに対しては，NAT テーブルによるアドレス変換を先に実行し，次に PAT テーブルでポート番号を変換することにより EN へ送信する．

このように NTS ルータでは natd の NAT テーブルと NTS ルータモジュールの PAT テーブルを組み合わせることにより，NTS ルータとして必要な NAT テーブルを実現している．なお，NAT テーブルの有効時間は UDP が 300 秒，コネクション確立後の TCP が 86,400 秒（24 時間） であり，PAT テーブルにも同様の値を適用する．

5. 評　　価

試作システムを実装し，EN と IN が通信を行う場合の動作確認と性能測定を行った．EN から IN に対して FTP 接続を行い，問題なくファイル転送が行えることを確認した．また，NTS ルータ配下の複数の IN に対して同時に HTTP 通信を開始できることを確認した．

提案方式の具体的な利用状況として，ユーザが携帯端末を用いて家庭内のサーバにアクセスすることが考えられる．また，この技術を用いることにより，無線 LAN のホットスポットをプライベートアドレスで構築することが可能になり，プロバイダがサービスを提供することなどが想定できる．

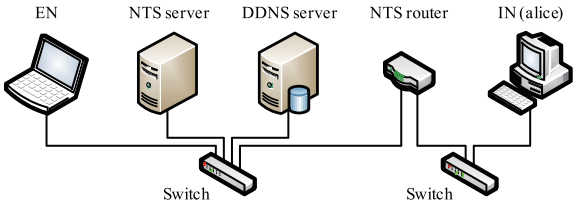


図 8　実験システムの構成
Fig. 8　Configurations of experiment systems.

5.1 評 価 内 容

提案システムの有用性，妥当性を判断するために，通信開始時にかかるオーバヘッドおよびスループットが必要であると判断し，それらの測定を行った．通信開始時には通常の DNS 名前解決に比べ，NTS サーバと NTS ルータ間の通信が追加されるため，その分のオーバヘッドが増加する．これはすべてのアプリケーションで共通の動作である．また，通信時のスループットについては，NTS ルータにおいて PAT テーブルに基づくポート変換処理が追加されたため，このための処理負荷が増加する．実験では，TCP と UDP について，提案方式を適用した場合としない場合の比較を行った．上記実験により，提案システムを適用しても TCP/UDP 上で実現されたアプリケーションが従来のシステムとほぼ同等の性能が出せることを示す．

実験システムの構成を**図 8** に示す．各装置の仕様はすべて共通で，CPU が Pentium4 3.0 GHz，メモリが 512 MB であり，100BASE-TX でスイッチに接続した．

5.2 測 定 結 果

提案方式の通信開始時に発生するオーバヘッドを明らかにするために，通信が開始されるまでの時間をネットワークアナライザ Wireshark[9] を用いて測定した．**図 9** に通信開始時のオーバヘッド測定結果を示す．EN が DNS クエリを送信してから，NTS ネゴシエーションを経て DNS 応答を取得するまでの時間は 1,841.8 μs であった．このうち，NTS サーバと NTS ルータ間のネゴシエーションに要した時間は 265.2 μs で，NTS サーバにおける NTS メッセージ処理時間は 360.2 μs であった．

実際のシステムにおいては，通常の DNS 名前解決時間に NTS サーバと NTS ルータ間の 1 ラウンドトリップ時間と NTS メッセージ処理時間を加算した値が通信開始時のオーバヘッドとして見積もることができる．NTS サーバに該当する DNS キャッシュが無ければ，

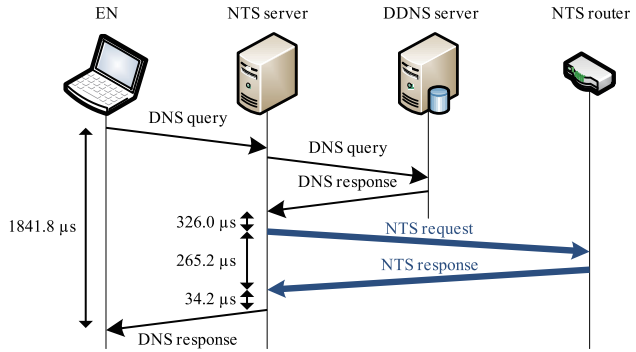


図 9　Wireshark による通信開始時のオーバヘッド測定値
Fig. 9　Measurement results of initial communication with Wireshark.

上位 DNS サーバにクエリを転送したり，反復クエリを使用してルート DNS サーバから順に名前解決を行う．日本国内では多くのルート DNS サーバからの応答時間は 300 ms 以下であり[10]，さらに TLD（Top Level Domain）サーバ以下それぞれの経路における通信遅延が加算される．このようなことを考慮すると，今回追加される通信開始時のオーバヘッドは実用上ほとんど問題ないといえる．

次に，NTS ルータにおける PAT テーブルの変換処理がスループットに与える影響を明らかにするために，Netperf[11] を用いて EN から IN への TCP/UDP スループットを測定した．比較のために提案方式を実装しない環境として，通常の natd でポートフォワーディング（静的 NAT）の設定を行い，EN 側から IN 側へ通信を行った場合のスループットと比較した．スループットの測定時間は 10 秒間とし，測定結果はいずれも 10 回試行の平均値である．**表 1** にスループット測定結果を示す．NTS 実装時，未実装時のスループットは TCP/UDP とも，どのメッセージサイズにおいても，両者の間には有意差が認められなかった．PAT テーブルに基づくポート変換の処理は，全体の処理に比べるとほとんど無視できることがわかる．

5.3　セキュリティに関する考察

ローカルネットワークは NAT により内部の IP アドレスが隠蔽されていたため，外部から特定の IN を標的とした攻撃を防止できるという効果がある．そのため企業ネットワークでは，NAT は簡易的なセキュリティ対策の為に欠かせないという側面もある．今回のような NAT 越え技術により，IN が脅威にさらされる可能性がある．

表 1　Netperf によるスループット測定値
Table 1　Measurement results of throughput with Netperf.

Message size (Bytes)	TCP (Mbps)		UDP (Mbps)	
	NTS	NAT	NTS	NAT
64	94.1	94.1	49.3	49.3
128	94.1	94.1	66.0	66.0
256	94.1	94.1	79.6	79.6
512	94.1	94.1	88.9	88.9
1024	94.1	94.1	94.4	96.4
1472	94.1	94.1	96.4	96.4

NTS：提案方式による NAT 越え通信
NAT：ポートフォワーディングによる NAT 越え通信

提案方式では IN が外部からの通信を許容する場合，DDNS サーバに名前と IP アドレスを登録をする．これは自分自身を外部に公開しているのと同じであり，IN がグローバル IP アドレスを取得した場合と同様の状況になる．従って，今回の提案により新たに発生する脅威ではない．

NTS ルータは提案方式に関する処理を行う前に，ファイアウォール処理を必ず実行する．通常のシステムと同様に適切なフィルタリングを設定することが重要である．また，NTS ルータの PHL に IN に対するアクセス制御の仕組みを導入することにより，アクセスが許可されていない IN に対して NTS ルータが外部からの指示で NAT テーブルを生成することを防止できる．更に NTS ルータ管理者は特定の NTS サーバからの通知のみ許可するように設定しておくことにより，不正アクセスなどの脅威から IN を保護することができる．

6. ま と め

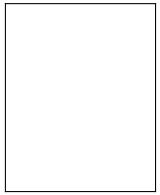
本論文ではユーザ端末の改造が不要な NAT 越えを実現する方式 NTSS を提案した．提案方式では EN の通信開始に先駆けて NTS サーバと NTS ルータが協調し，NTS ルータがオンデマンドに NAT テーブルを生成することにより NAT 越え通信を可能にする．各端末間の通信はエンドエンドで行うことができ，今後のユビキタスネットワーク社会に有益なシステムと考えられる．

プロトタイプシステムの実装を行い，複数の内部端末と同時に通信できることを実証し，性能測定により提案方式によるオーバヘッドは実用上問題ないことを示した．本提案システムは DNS の実装に依存しているため，DNS キャッシュに関するセキュリティ問題などを今後考察する必要があると考えられる．

参 考 文 献

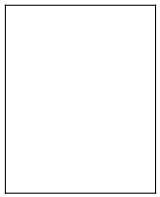
1) Juniper Networks: *Concepts & Examples ScreenOS Reference Guide: Part 8, Address Translation* (2009). http://www.juniper.net/techpubs/software/screenos/screenos6.3.0/630_ce_AddressTranslation.pdf
2) Rosenberg, J., Weinberger, J., Huitema, C. and Mahy, R.: STUN - Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs), RFC 3489, IETF (2003).
3) Rosenberg, J., Mahy, R., Matthews, P. and Wing, D.: Session Traversal Utilities for NAT (STUN), RFC 5389, IETF (2008).
4) Rosenberg, J., Mahy, R. and Matthews, P.: Traversal Using Relays around NAT (TURN): Relay Extensions to Session Traversal Utilities for NAT (STUN), Internet-draft, IETF (2009). http://tools.ietf.org/id/draft-ietf-behave-turn-16.txt
5) UPnP Forum: *Internet Gateway Device (IGD) Standardized Device Control Protocol V 1.0* (2001). http://www.upnp.org/standardizeddcps/igd.asp
6) Turányi, Z., Valkó, A. and Campbell, A.: 4+4: An Architecture for Evolving the Internet Address Space Back Toward Transparency, *ACM SIGCOMM Computer Communication Review*, Vol.33, No.5, pp.43–54 (2003).
7) 鈴木秀和，宇佐見庄五，渡邊　晃：外部動的マッピングにより NAT 越え通信を実現する NAT-f の提案と実装，情報処理学会論文誌， Vol.48, No.12, pp.3949–3961 (2007).
8) Ng, T., Stoica, I. and Zhang, H.: A Waypoint Service Approach to Connect Heterogeneous Internet Address Spaces, *Proc. USENIX Annual Technical Conference*, pp.319–332 (2001).
9) Wireshark Foundation: Wireshark: Go deep. http://www.wireshark.org/
10) 関谷勇司，長健二朗，加藤　朗，村井　純：基準 DNS サーバを利用した DNS のパフォーマンス測定並びに評価手法に関する研究，電子情報通信学会論文誌 (B)， Vol.87, No.10, pp.1542–1551 (2004).
11) Jones, R.: Netperf: a network performance monitoring tool. http://www.netperf.org/netperf/NetperfPage.html

(平成 21 年 9 月 21 日受付)
(平成 22 年 6 月 4 日採録)



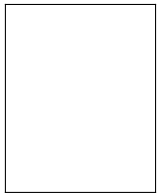
宮﨑　悠（正会員）

2007 年名城大学理工学部情報科学科卒業．2009 年同大学大学院理工学研究科情報科学専攻修了．同年 KDDI 株式会社入社．運用統括本部所属．修士（工学）．



鈴木　秀和（正会員）

2004 年名城大学理工学部情報科学科卒業．2006 年同大学大学院理工学研究科情報科学専攻修了．2009 年同大学院理工学研究科電気電子・情報・材料工学専攻博士後期課程修了．2008 年日本学術振興会特別研究員．2010 年より名城大学理工学部情報工学科助教．ネットワークセキュリティ，モバイルネットワーク，ホームネットワーク等の研究に従事．博士（工学）．電子情報通信学会，IEEE 各会員．



渡邊　晃（正会員）

1974 年慶應義塾大学工学部電気工学科卒業．1976 年同大学大学院工学研究科修士課程修了．同年三菱電機株式会社入社後，LAN システムの開発・設計に従事．1991 年同社情報技術総合研究所に移籍し，ルータ，ネットワークセキュリティ等の研究に従事．2002 年名城大学理工学部教授，現在に至る．博士（工学）．電子情報通信学会，IEEE 各会員．