ネットワークの可視化による NAT 越え

NAT Traversal by i-Path Network Transparency

戸部 和洋† 下田 晃弘‡ 後藤 滋樹†‡

†早稲田大学 理工学部 コンピュータ・ネットワーク工学科
‡早稲田大学 基幹理工学研究科 情報理工学専攻

概要

通信相手ごとに別のポート番号を割り当てる NAT (Symmetric NAT) を Hole Punching で越えるには，次に割り当てられるポート番号を予測する必要がある．従来の技法には，自分側の NAT と相手側の NAT の間で IP パケットが破棄されるように TTL の値を調節する手法がある．しかし，経路上に複数の NAT が存在すると，最適な TTL の値を予測することは難しい．本論文では，通信パスに関する情報をエンドノードに提供するフレームワーク (i-Path) を利用してルータが NAT の情報を公開し，その情報を活用することで，Hole Punching による NAT 越えの問題を解決できることを示す．

1 i-Path Project

i-Path Project [1] は，in-band クロスレイヤ方式を拡張して，通信パスに関する情報をエンドノードに提供することで，ネットワークを可視化するフレームワークである．エンドユーザは既存の traceroute の機能に加え，通信パス上に存在するルータの位置情報やスループットといった固有情報を取得することができるようになる．そして，i-Path では開示ポリシを尊重しており，ISP と送受信者が合意した情報だけを開示する機構をもつ．

2 NAT 越え (NAT Traversal)

Network Address Translator (NAT) の内側のホストの IP アドレスは，外側のホストにはわからない．また，NAT はインバウンドの開始パケット (initiation packet) を破棄する性質をもつ．そのため，通常は NAT を越えて Peer-to-Peer (P2P) 通信を行うことができない．これに対し，NAT を越えて NAT の内側のホストと直接通信する NAT 越えの手法が研究・開発されている．

2.1 Hole Punching

Hole Punching とは，(1, 2) NAT の内側からパケットを送信して NAT に穴（ポート）を空け，(2, 3) そのポートをめがけて外側からパケットを送信することにより，NAT 越えを実現する技術である．UDP Hole Punching のシーケンスを図 1 に示す．

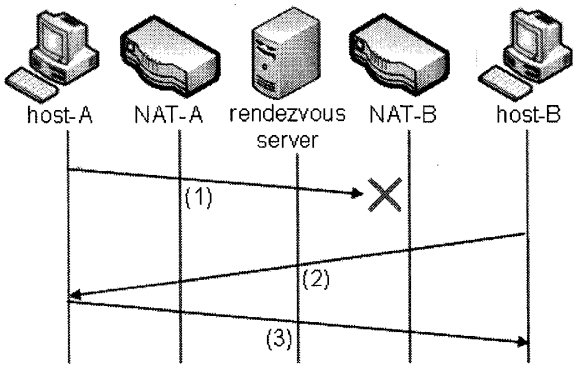

図 1　UDP Hole Punching のシーケンス

2.2 ポート予測 (Port Prediction)

Hole Punching には，相手ホストが NAT に割り当てられた IP アドレスとポート番号を知る必要がある．このため，(1) の通信に先立って，host-A, B は rendezvous server と通信を行う（この通信を (0) とする）．rendezvous server はそのパケットから，host-A, B が NAT-A, B に割り当てられた IP アドレスとポート番号を取得し，host-A, B に通知する．しかし，Symmetric NAT [2] と呼ばれる NAT は，通信相手が異なると新しいポート番号を割り当てる．そのため，(0) と (1) では，NAT は異なるポート番号を割り当てる．その結果，(2) や (3) のパケットは NAT に破棄されてしまう．

ところが，実際にはポート番号の割り当てが規則的で，次のポート番号を予測できることがある．UDP Multi Hole Punching [3] は，2 台のサーバとの通信により

ポート予測を行い，大量のパケットを送信することで，Symmetric NAT [2] を越えられる確率を高めている．

2.3 Low TTL Value Determination

UDP Multi Hole Punching [3] や NATBLASTER [4] は，パケットの TTL を低い値にして送信することで，自分側の NAT を越えて相手側の NAT に到達する前にパケットが破棄されるようにする．例えば，図 2 のネットワークでは，初期 TTL を 2 にしてパケットを送信すると，2nd hop のルータで時間切れ (Time Exceeded) となり，パケットが破棄される．ここで，TTL の初期値をいくつにすればよいのかという問題が生じる．NAT は共同住宅や ISP に配置されて多段になることがある．

この問題に対しては次のような対策が考えられる．NATBLASTER [4] で提案されているように，traceroute/tracert を用いて通信経路のホップ数を調べられる．また，各ホップの IP アドレスを見ることにより，そのルータが NAT をしているか調べられる．NAT 内部のネットワークでは，プライベートアドレスなど特別な IP アドレスが使われることが多いからである．

しかし，traceroute で送信する UDP パケットは，そのポート番号がほかのアプリケーションで使用されていたり，パケットフィルタリングが行われていたりして応答が得られないことがある．また，ICMP Echo Request を送信しても，ICMP Echo Reply を返さないように設定された機器や，ICMP パケットを処理しない NAT が存在するため，応答が得られないことがある．

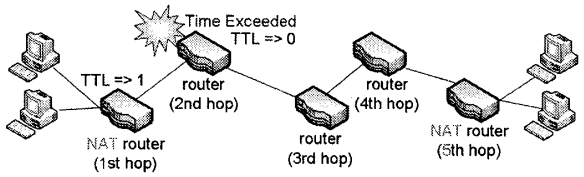

図 2 初期 TTL を 2 にした場合

3 NAT 情報の公開

NAT がもつポート番号の割り当てアルゴリズムなどの情報は，UDP Multi Hole Punching [2] がするようにパケットを送信してその挙動を調べれば推測できる．そして，この情報が第三者に知られても不都合が生じることはない．ルータが NAT を行っているかどうかという情報についても同様である．2.2 節と 2.3 節で述べたように，これらの情報を調べるためには手間がかかり，正確に判定できないこともある．そこで，i-Path で経路上のルータがこれらの情報や TTL の値をエンドノードに提供することで，Hole Punching にかかる時間やリソースを節約でき，信頼性も向上することができる．

4 考察

4.1 UPnP との比較

UPnP (Universal Plug and Play) に対応した NAT ルータは，IP アドレスとポートの関連づけの取得や，ポートフォワーディングの設定を端末側から行える．しかし，UPnP のアクセスはローカルネットワーク内の NAT に限定される．また，UPnP には認証機構がない．よって，多段の NAT に対応し，公開ポリシを柔軟に設定できる本提案は有意義である．

4.2 エンドツーエンド NAT との比較

エンドツーエンド NAT [5] は，積極的に NAT の存在と様態を知らせ，エンドノードが NAT の動作を補完することで，エンドツーエンド透過性を実現する．一方，本提案ではルータが NAT の情報を公開することで，エンドノードによる NAT 越えを補完する．そのため，エンドツーエンド NAT ではエンドノードのプロトコルスタックを修正する必要があるが，本提案では NAT 越えが必要なアプリケーションだけを修正すればよい．

謝辞

本研究は情報通信研究機構の委託研究「新世代ネットワークサービス基盤構築技術に関する研究開発」の一環として行われた．

参考文献

[1] D. Mochinaga, K. Kobayashi, S. Goto, A. Shimoda and I. Murase, Collecting inside information to visualize network status, APAN Network Research Workshop 2009, pp.1–4, August, 2009.

[2] J. Rosenberg, J. Weinberger, C. Huitema and R. Mahy, STUN - Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs), RFC 3489, March, 2003.

[3] Y. Wei and S. Goto, New Method for Symmetric NAT Traversal in UDP and TCP, APAN Network Research Workshop 2008, August, 2008.

[4] A. Biggadike, D. Ferullo, G. Wilson and A. Perrig, NATBLASTER: Establishing TCP connections between hosts behind NATs, Proceedings of ACM SIGCOMM Asia Workshop, April, 2005.

[5] 太田 昌孝, 森岡 仁志, 藤川 賢治, エンドツーエンド NAT, 信学技報, vol. 109, no. 137, pp. 1–6, July, 2009.