エンドツーエンドNAT

太田昌孝
東京工業大学情報理工学研究科
mohta@necom830.hpcl.titech.ac.jp

IPアドレスが足りない！！

- それでもIETFなら、、、IETFならきっと何とかしてくれる！？
  - いまだに、何ともなっていない、なりそうもない
    - NATによるアドレス節約
      - エンドツーエンドインターネットを破壊
      - いろんなプログラムが、まともに動作しない
    - IPv6によるアドレス拡張
      - 実運用を考えない政治的妥協の産物
      - オプションヘッダ、PMTUD、Stateless AC、Link Local Address等有害無益な機能を盛込み過ぎ

SALTZER等の原論文でのエンドツーエンド論法

http://groups.csail.mit.edu/ana/Publications/PubPDFs/End-to-End%20Arguments%20in%20System%20Design.pdf

- The function in question can completely and correctly be implemented only with the knowledge and help of the application standing at the end points of the communication system. Therefore, providing that questioned function as a feature of the communication system itself is not possible. (Sometimes an incomplete version of the function provided by the communication system may be useful as a performance enhancement.)

背景

- エンドツーエンド論法によると、
  - NAT can completely and correctly be implemented only with the knowledge and help of the application standing at the end points of the communication system
    - エンドホストの知識と助けを利用していない今のNATは、不完全で不正確でエンドツーエンド透過性をもたない
- ならば、逆は成り立つのか？
  - With the knowledge and help of the application standing at the end points of the communication system
    - Can NAT be implemented completely and correctly?

エンドツーエンドNAT

― NATの存在を端に積極的に見せる ―

- プライベート網内の端末にNATGWの知る
  - 各端末で共有するパブリックアドレス
  - 各端末に割り当てたポートの範囲
  - NATGWとの通信方法（アドレス、ポート）

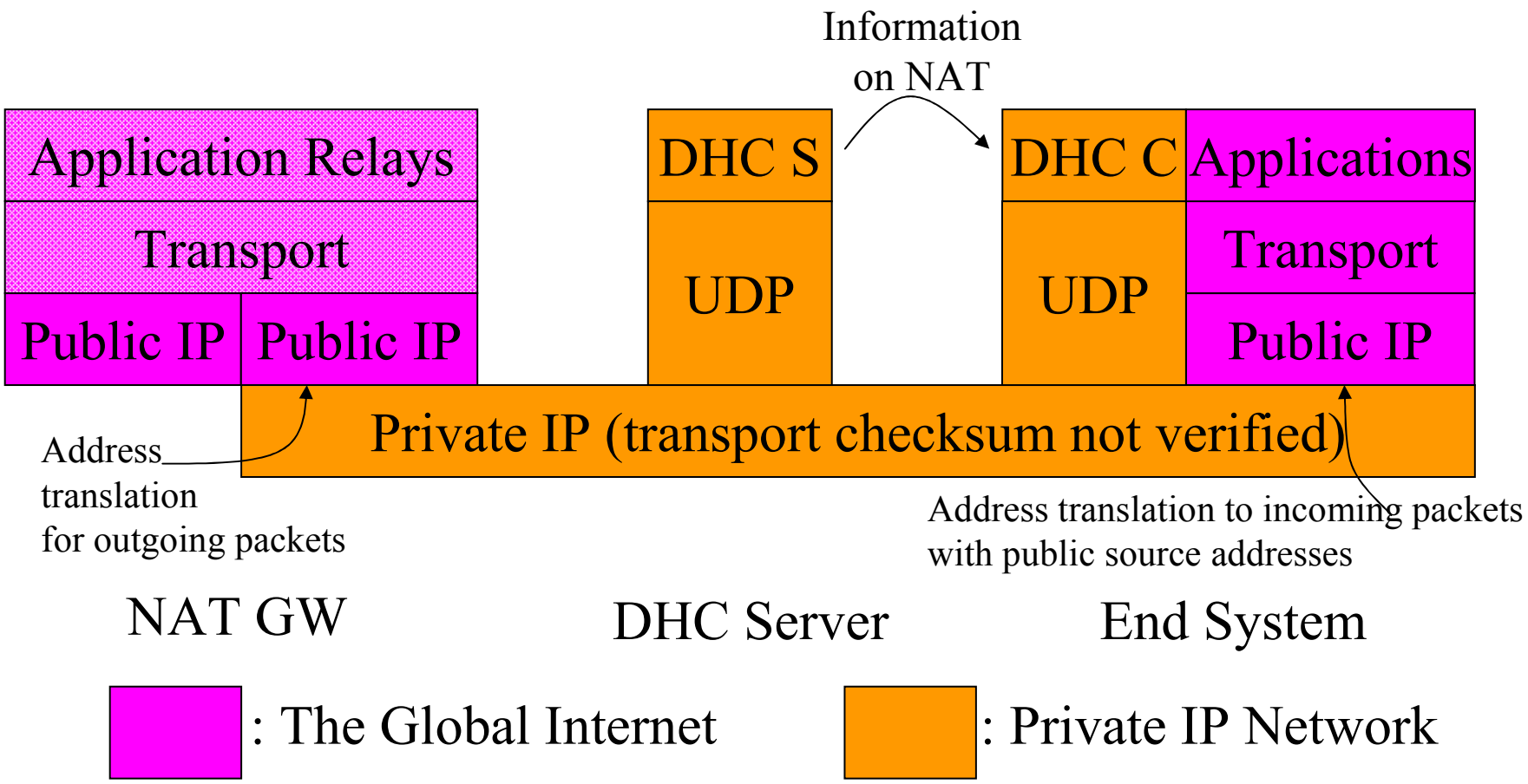
等の情報を、DHCPやPPP等で通知

- 各端末は
  - 自らの知識により、NAT動作が完全かつ正確なものになるように、補完する

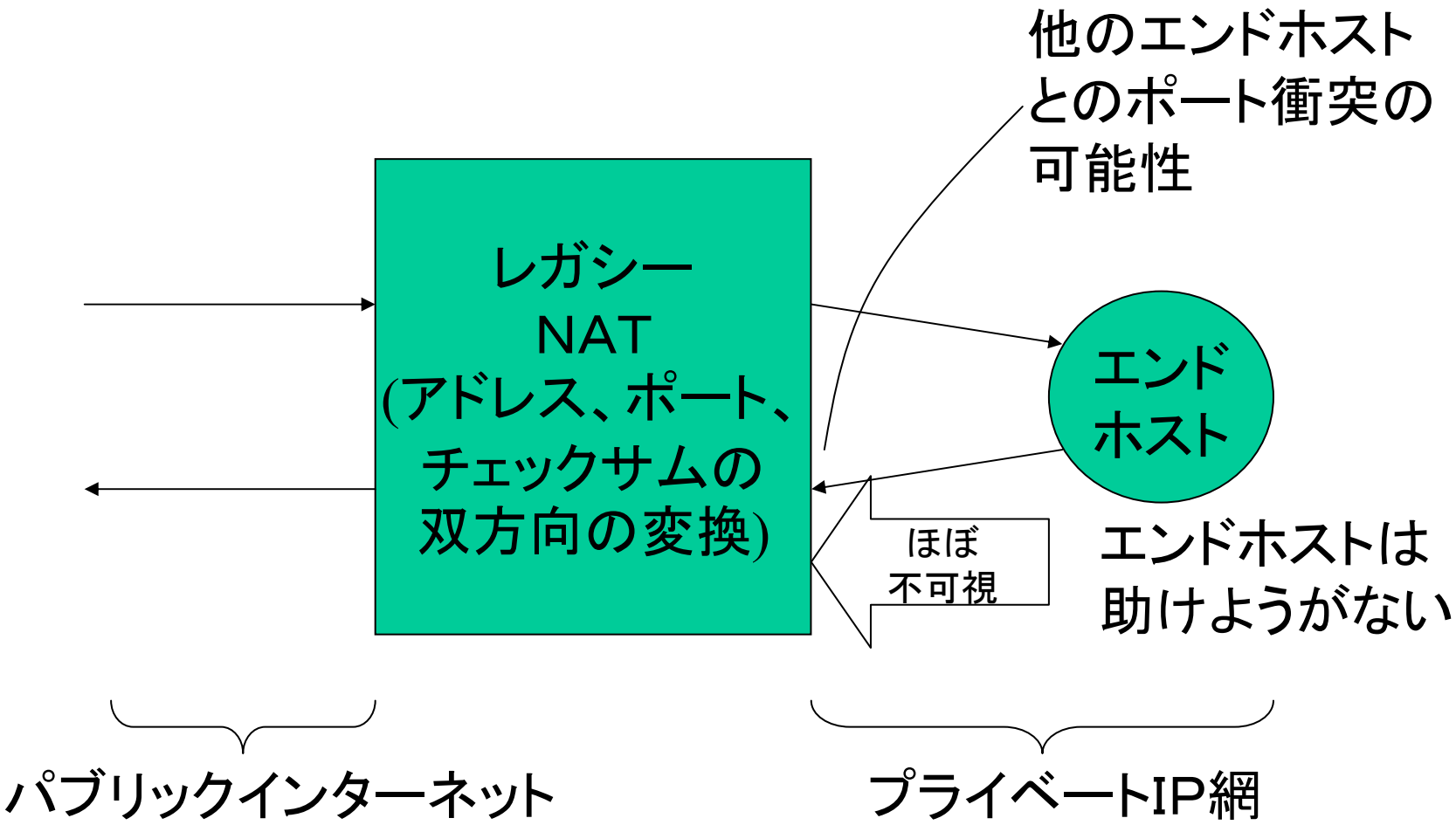
エンドツーエンドNATの動作

- NATゲートウェイ
  - 受信者ポート番号により、パケットの受信者アドレスをパブリックアドレスから変換
    - ポートやトランスポートチェックサムは変換せず
- NAT背後のエンドホスト
  - 受信者アドレスをパブリックアドレスに逆変換
    - トランスポートチェックサムは正しくなる
  - 送信者ポート番号を、エンドホストに割り当てられたものに制限

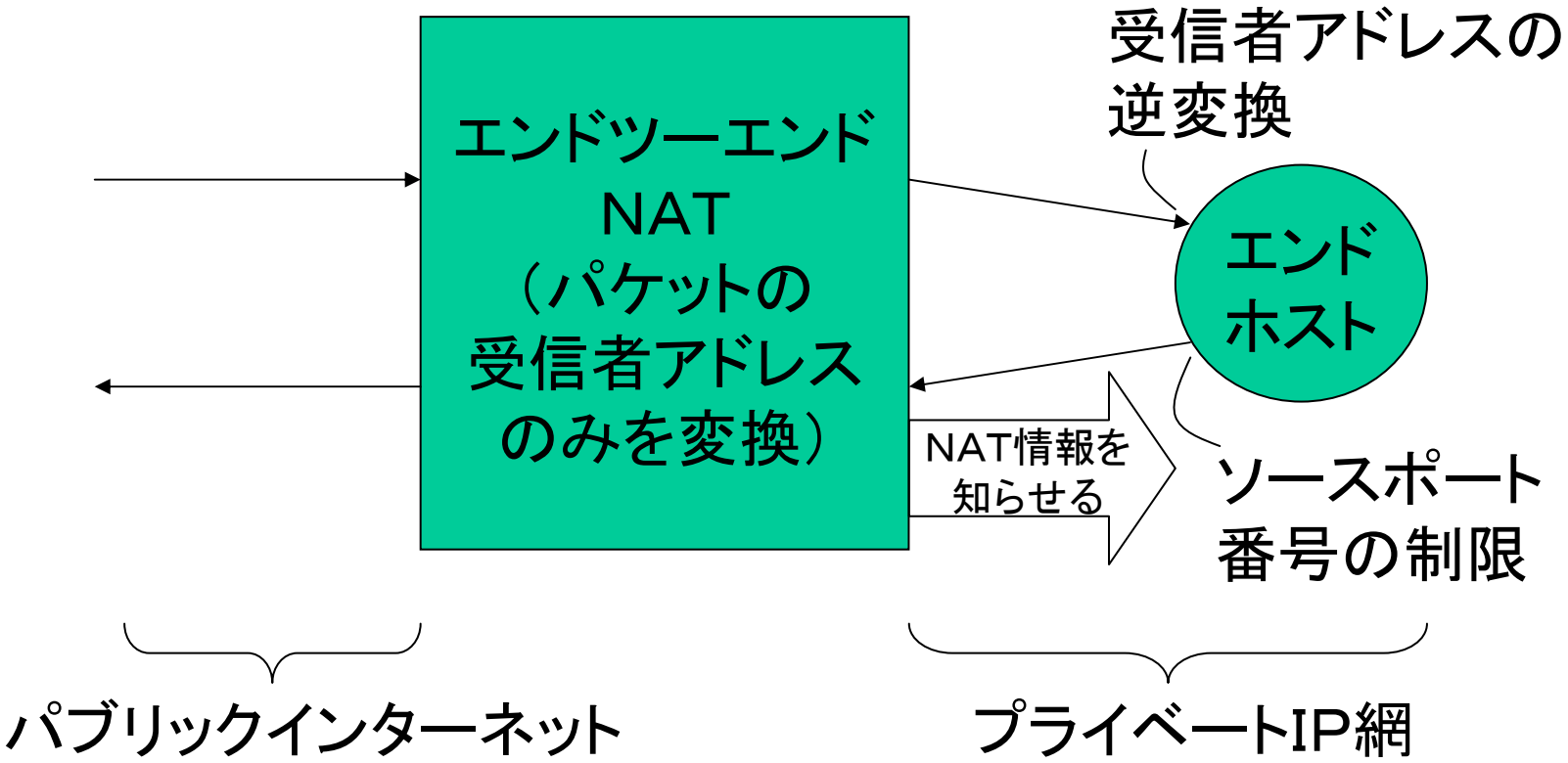
エンドツーエンドNATの レイヤー構造



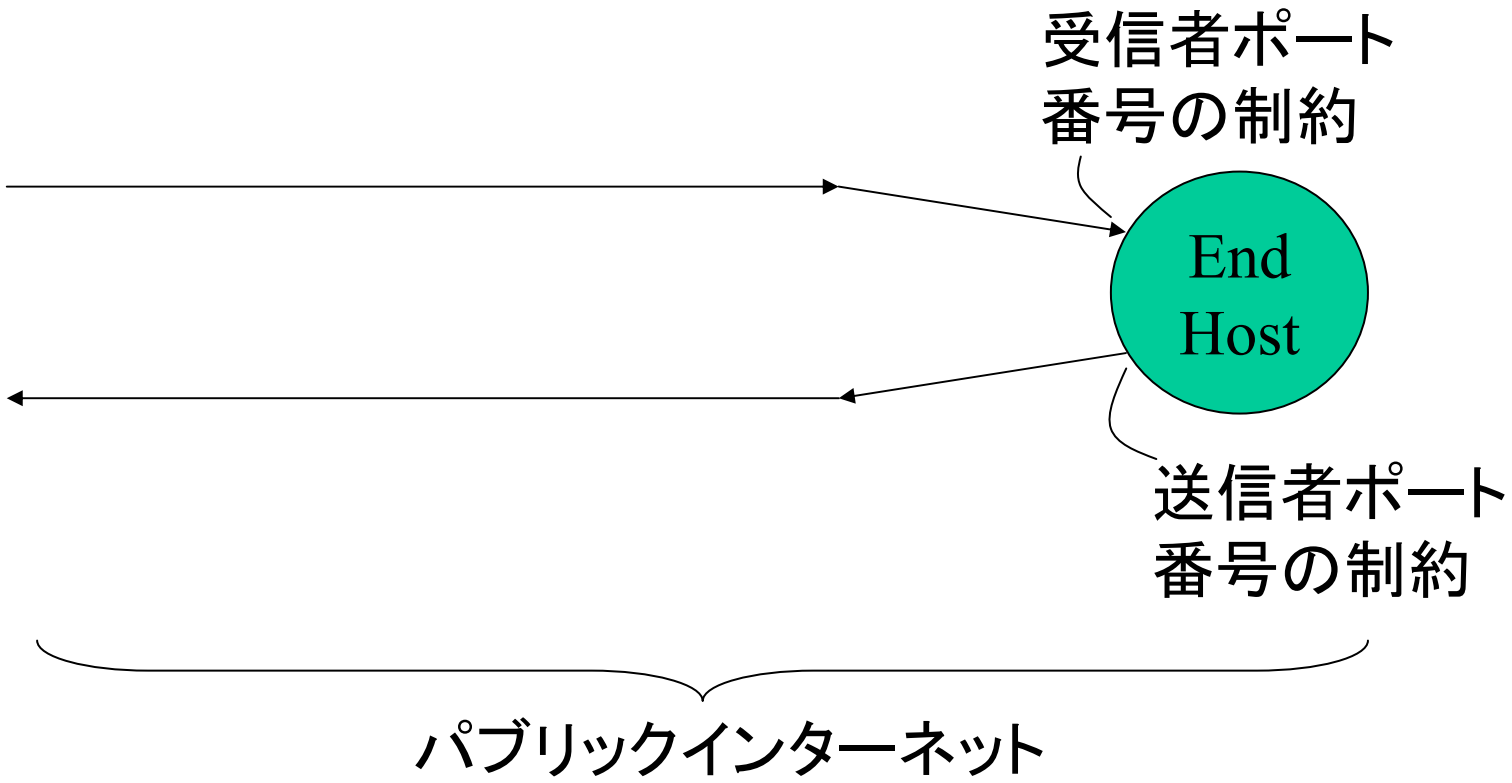
レガシーNAT


他のエンドホスト
とのポート衝突の
可能性
レガシー
NAT
(アドレス、ポート、
チェックサムの
双方向の変換)
エンド
ホスト
ほぼ
不可視
エンドホストは
助けようがない
パブリックインターネット
プライベートIP網

エンドツーエンドNAT



エンドツーエンドNAT背後のエンドホストはインターネット直結と等価



エンドツーエンドNATの性質

- 完全なエンドツーエンド透過性
  - ポート番号か同等物さえ存在すれば
  - ftpのポートコマンドも自然にNATを超える
- 多段ネスティング可能
- コンパチビリティも豊富
  - レガシーNAT、ICMP、(ポート番号も含め)DNS逆引き、マルチキャスト、モビリティ(ポート単位の移動のため要拡張)、IPSEC、、、

スタティックNATと
ダイナミックNAT

- スタティックNAT
  - 各端末に固定したポート範囲を割り当て
    - ポート数が多ければ（数百？）、これで十分
      - 端末は自己に割り当てられたポートのみをソースポートとして使う。返事が貰えないので、偽装は無理。
- ダイナミックNAT
  - 各端末は、ポート番号を随時NATGWに要求
    - NATGWのポート割当状態は、端末主導で更新
      - タイムアウト、複数GW間整合性等の問題は解消

固定ポートでのE2ENATとポートフォワーディングとの違い

- ポートフォワーディングでは
  - 一部のポートを特定の端末に固定割当
    - 旧来のNATが、トランスポート層で中継
    - スタティックNAT同様、端末はサーバ動作可能？
      - 実は透過性は無く、クライアントすらまともに動かない
- E2ENATでは
  - 端末の助けにより、完全なE2E透過性を実現

エンドツーエンドNATと ポート番号

- 多くのアプリでは、デフォールト以外のポート番号をURLで陽に指定可
- E2ENATはほぼIP層だけで動作するが
  - 受信者ポート番号はIPヘッダの外にある
    - 純トランスポートプロトコルでは、IPヘッダ直後16ビットが送信者ポート、次16ビットが受信者ポート
    - ICMPの仕様から、ICMP以外は、送信者ポート番号は、IPヘッダ直後の8バイトに含まれるはず
      - ポート番号は、IPヘッダ直後8バイト内の2バイト境界間の16ビットと決め打ちしてもよさそう（IPSECも対応可）

エンドツーエンドNATと ICMPエラー

- ICMPエラーは、エラーを起こした内部パケットの送信者ポートによりアドレス変換
- ICMP　HOST　UNREACHは
  - パブリックアドレスを共有する他ホストに影響
  - ソフトエラーなので、気にしない
    - TCPは接続を切らない
    - 気を利かせたつもりでPORT　UNREACHに変換すると、ハードエラーなので悲惨

エンドツーエンドNATとICMPエコー

- ICMPエコーリクエストは
  - IDとSEQ＃をそれぞれ送信者と受信者のポート番号と看做し
    - IDを制約、SEQ＃でアドレス変換
- ICMPエコーリプライは、逆
  - IDとSEQ＃は、ICMPエコーリクエストからコピーされる
  - IDを受信者ポートと看做しアドレス変換すれば、エコーリクエストの送信者に届く

エンドツーエンドNATと IPSEC

- AHもESPも32ビットSPIをIPヘッダ直後の8バイト以内に持つ
- SPIの値を決める際、前半16ビットを送信者が指定し、後半16ビットを受信者が指定することとすれば
  - 送信者ポート番号、受信者ポート番号として利用可能

エンドツーエンドNATと アプリケーションリレー

- DNS、SMTP、HTTP等は、NATGWのデフォールトポートでリクエストを受け、リクエスト中の情報（ドメイン名等）で、リクエストを端末に振り分け可
  - HTTP：のURLでのポート指定は不要に
  - DNSやSMTPのポートはNSとMXに内包され（URLで指定不可）、プライベート網内にサーバを置くには（置けなくても、ほとんど困らないが）、アプリケーションリレーは必須

エンドツーエンドNATと エンドホストのアドレス使い分け

- エンドホストは、自分のパブリックアドレスとプライベート網のアドレスを知っている
  - プライベート網へのパケットのソースアドレスはプライベートアドレスで
  - それ以外のアドレスへのパケットのソースアドレスはパブリックアドレスで
  - 自分のパブリックアドレスへのパケットは、自分のポート番号じゃなければ出力する

エンドツーエンドNATと非対応端末

- NATGW背後のE2ENAT非対応端末は
  - DHCP等によるアドレス割り当ては受けても
    - NAT情報は理解できない
- 非対応端末が出したパケットに対して
  - NATGWは旧来のNATとして対応してもよい
    - UPnP機能等もあってもよい
- E2ENAT対応端末との区別は
  - ソースアドレスで判別可

エンドツーエンドNATのネスト

- E2ENATGWはネスト可能
- ISPからスタティックNATで多数（数百？）のポート番号を割り当てられた顧客は
  - 一部をサーバで固定的に利用
  - 一部はダイナミックNATGWの外側に割当
    - ダイナミックNAT背後にネストしたプライベートネットワーク内の多数の端末で、ポート番号をダイナミックに共有

エンドツーエンドNATと逆引き

- 共有アドレスは普通に逆引き可能

  www.example.com A 208.77.188.166

  166.188.77.208.in-addr.arpa PTR www.example.com

- ポート別の逆引きは以下のように可能

  p1.example.org CNAME www.example.com

  1.0. 166.188.77.208.in-addr.arpa PTR p1.example.org

  p2.example.org CNAME www.example.com

  2.0. 166.188.77.208.in-addr.arpa PTR p2.example.org

  – PTRがCNAMEを指すことは、PTRから先の自動参照の懸念（RFC1034）はないので、問題ではない

エンドツーエンドNATと マルチキャスト

- マルチキャストアドレスは、内外で共通
- プライベート網内の端末が送信する場合
  - マルチキャスト経路制御にはソースアドレスへの経路が影響するので
    - マルチキャストパケットはNATGWが出すべき
    - 端末は、送信すべきパケットを、IP over IPで、NATGWへ転送
      - PIMの仕組みでも使えばよい

エンドツーエンドNATとモビリティ

- ホームアドレスがNAT背後にいる場合
  - NAT情報をMHに設定（静的設定で十分）
  - ホームNATGWとの通信は、HAが中継
- MHがNAT背後にいる場合
  - フォーリンアドレスとホームアドレスで、使えるポートは一般には一致しない
  - HA－＞MHのトンネルをIP over UDP over IPにすれば、解決
    - フォーリンポートは一個で十分（アドレス節約）

実装

- NetBSD5．0ベース、静的のみ
- 本質的改造は、
  - アドレス変換と逆変換のため
    - 端末のip__input．cへの数行の追加
    - GWのip__input．cの数十行の追加
  - ソースアドレスとソースポートの制限のため
    - 端末とGWのin__pcb．c等の数百行の追加
    - 端末とGWのip__output．cの数行の追加
      - NICにトランスポートチェックサム計算をやらせない

エンドツーエンドNATのデモ

- 時間とネットワーク環境の許す限り、、、

エンドツーエンドNATと アドレス分配ポリシー

- E2ENATは、現状のインターネット環境をエンドツーエンド透過性も含めほとんど全て保ちながら、アドレスを大幅に節約
- E2ENATをアドレス分配の前提にすべき
  - ISPの労力は？
    - どう考えても、IPv6とのデュアル運用より少ない
    - 特に、アドレスが暗記できるのは、非常に重要
- クラスEアドレスも利用すべき

エンドツーエンドNATへの ISPの対応

- 旧割当はそのままに、新アドレスについて
  - スタティックNATGWを配備
    - 必用に応じて、アプリケーションGWを配備
  - 固定アドレスを渡していない場合
    - DHCPで（一般には毎回異なる）アドレスとポート番号範囲を顧客に渡す
  - 固定アドレスを渡している場合
    - DHCPや書面で、顧客に応じたアドレスとポート番号範囲を顧客に渡す

エンドツーエンドNATと クラスEアドレス

- クラスEアドレスは、長持ちしない！？
- E2ENATによるアドレス節約前提なら
  - 移行期間の後、クラスEアドレスをユニキャストに使う意味はある
    - E2ENAT対応には端末の改造が必須なので
      - 同時にクラスE対応にすればよい
    - ISPやルータや既存の端末が対応しないと相互接続性が、、、
      - IPv6対応よりは、はるかに簡単（特にISPやルータ）

エンドツーエンドNATと プリフィックス

- 大域経路表に／24より長いのが増える？
  - 1600万あれば十分（というか多すぎ）
- どのみち、IPv6では、大域経路表プリフィックス数の抑制の試みは崩壊
- IPで時間を稼いで
  - エンドツーエンドマルチホーミングを実現
  - 新世代IPに、乞う御期待

エンドツーエンドNATと エンドユーザー

- E2ENATの導入によりエンドユーザーは
  - サーバーもクライアントも今と同様に動作
  - IPv6対応は不要に
    - アドレス（とポート）は、普通の人でも暗記可能
  - httpのURLではポート指定不要
  - 既存ユーザーはそのままで、新規ユーザーは
    - （旧来のNAT環境が嫌なら）端末改造が必要
      - このまま旧来のNAT導入するより、遥かによい

エンドツーエンドNATと NIC

- エンドツーエンドNATを導入すると
  - 割振サイズは小さく
  - 割振速度は低下（／256で割振った場合）
  - IPv6対応は不要に

エンドツーエンドNATプロトコル詳細議論用のメイリングリスト

- 日本語(*=ja)、と英語(*=en)を用意
  - e2enat-* at mobile-broadband.org
- 参加方法
  - e2enat-*-ctl at mobile-broadband.orgに、
  - subscribe Your-Last-Name Your-First-Name

まとめ

- E2ENATは、現状のインターネット環境をエンドツーエンド透過性も含めほとんど全て保ちながら、アドレスを大幅に節約
- E2ENAT前提のアドレス管理により、IPアドレス空間は（クラスEも使えばなおさら）当分持つ
- IPv6？なにそれ？