2007年度前期
情報システム構成論2
第4回「次世代インターネット技術」

西尾 信彦
nishio@cs.ritsumei.ac.jp
立命館大学 情報理工学部

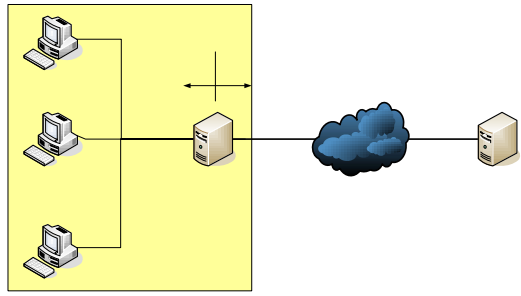
IPv4におけるNAT透過性問題

- End-to-end通信の必要性
  - どのホストとも直接接続したい
  - しかしホストの数は増えていく一方
  - 基本的にグローバルアドレスがなければ直接通信は不可
- IPv6かNATか？
  - グローバルアドレスがないホストはNATの内側
  - すべてがIPv6になるのはまだ時間がかかる
  - NATの壁を越えてEnd-to-endを実現させる技術が注目されている

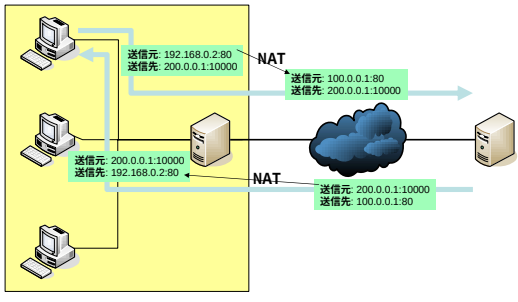
NATの役割

- 少数(通常1つ)のグローバルアドレスを付与され
- その内側にプライベートアドレスを付与したホスト群により構成されるプライベートネットワークを成立させる
  - 主に内側から外側への通信を実現する
  - そのため逆の通信は基本的に支援できない
- ファイアウォールなどセキュリティ機構は含まないが，
- 同一ハードウェアが同機構をもつことが多いため混同される
- NATしただけではネットワーク攻撃は防げない

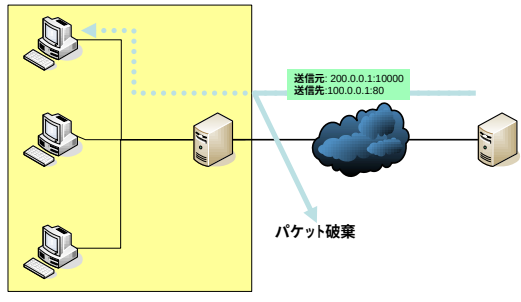
NATの基本構成



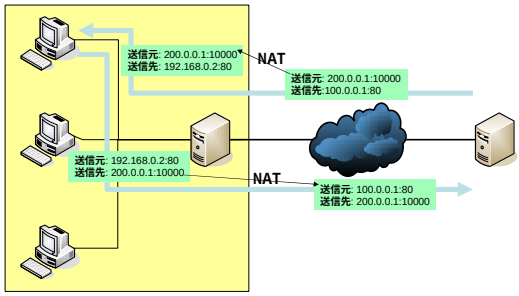
NATのアウトバウンド通信
通常の通信



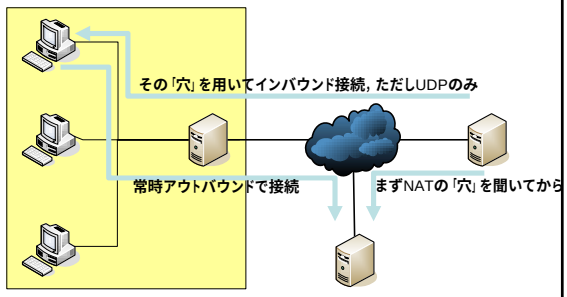
NATのインバウンド通信
パケット破棄される通信



NAT透過性の実現：ポートフォワード
NATエントリの静的設定



NAT透過性の実現：中継ホスト方式
STUN



NAT透過のIPsec

- ESPのセッション開始時のIKEがNATにはばまれるので
- トンネル方式は可能だが、トランスポート方式は通常不可能
- トンネル方式ではNATとIPsecのVPNホストを兼ねる
- トランスポート方式では、IETFによりIPsec-NAT-Tが提案
  - IPsecパケットをUDPでカプセル化
  - IKEプロセスをNAT上でネゴーシエイト

192.1

IPv6概要

- 128ビットアドレス
- 上位64ビットがネットワークプレフィックス
- 下位64ビットがホストアドレス
  - ネットワークにクラスがない
- 16ビットずつ：で区切って16進法で記述する
- 間に0が続く場合には：：と省略表記
  - fe80:0000:0000:0000:0000:0000:0000:0001
  - fe80::1

192.168.0.3

NAT

IPv6アドレスのスコープ

- 一つのインタフェースに2つ(3つ)のアドレス
- リンクローカルアドレス
  - ルータを越えない範囲で利用可能
  - MACアドレスを利用するのでネットワークプレフィックスさえわかれば設定が簡易化
- グローバルアドレス
  - 通常に配布されるend-to-endでグローバルに利用できるアドレス
- (既に廃止が叫ばれるが)サイトローカルアドレス
  - NATの弊害を繰り返す恐れがあるため

68.0.4

IPv6のアドレス方式

- ユニキャスト
  - 従来通り
- マルチキャスト
  - ブロードキャストはなくなった
- エニーキャスト
  - そのネットワークプレフィックスのどれかのホスト
  - もしくはそのホストのどれかのインタフェース

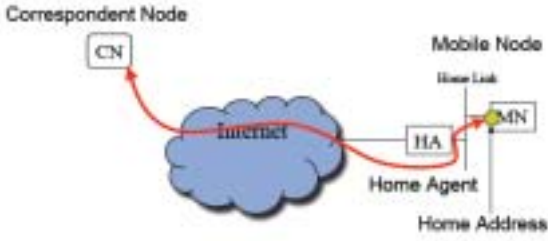
IPv6の近隣探索機能

- ICMPv6によるNDP: Neighbor Discovery Protocol
- Router SolicitationとRouter Advertisement
- Neighbor SolicitationとNeighbor Advertisement
  - リンクローカルのマルチキャストでクエリ
  - これらによりARPは必要なくなり、DHCPも必須ではなくなった
  - アドレスの多重割り当て確認も行なう

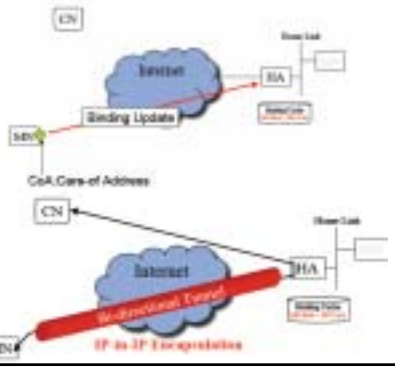
MobileIPv6

- ホストが異なったネットワーク間を移動するとき、
- 同じホストアドレスを使い続けても通信が可能にする技術
- IPv4時代からあったがIPv6で最適化された
  - IP-in-IPトンネリング
  - 三角ルーティングの解消
  - フォーリンエージェントの省略

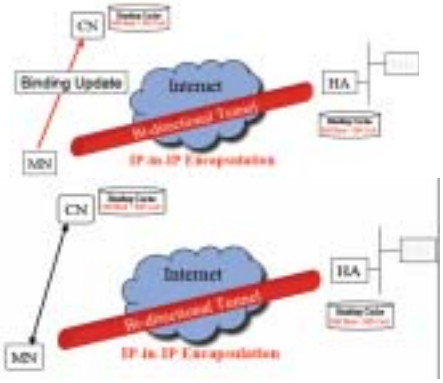
Mobile IPv6のメカニズム



At Foreign Link

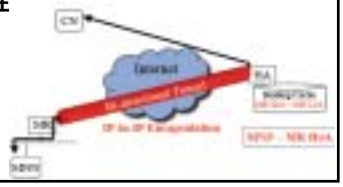


Route Optimization



さらにNetwork Mobility

- ネットワークが移動するとは
  - 多くのホストがグループになって移動する
  - 移動する車たち
  - 車の中の多数のコンピュータとセンサー
  - 多くの人(=携帯端末)を乗せた列車
- さらなる最適化の重要性
  - MobileRouteによる
  - NeMoから
  - Nested Mobilityへ



IPv6への移行

- デュアルスタック
  - IPv4とIPv6の両方をしゃべるホスト
- トンネリング
  - IPv6パケットをIPv4パケットにつつんで遠方のIPv6ネットワークまで届ける
- トランスレータ
  - NAT-PTやSIIT
  - End-to-endにはならない

IPv6トンネリング

- 自動トンネリングプロトコル
  - 6 to 4
  - ISATAP
- NAT環境でのトンネリング
  - Teredo

本当にIPv6に移行できるか？

- OS(プロトコル実装)、ルータ(ciscoの怠慢)、アプリケーション(キラーアプリの欠乏)などといわれるが
- 真の敵は
  - IPv6がIPv4の上位互換性をもっていないことか？
  - いつまでたってもIPv4への投資が減少できない

2007年度前期
情報システム構成論2
第5回「Ad-hoc NetworkとP2P技術」

西尾 信彦
nishio@cs.ritsumei.ac.jp
立命館大学 情報理工学部

Ad-hocネットワーク技術

講義では湧川隆次@慶應SFC氏によるものを多数利用しています
そちらを参照してください

P２Pネットワーク技術

P2Pとは？

Peer to peerの省略形

peer
【名】<社会的･法的に>地位の等しい人、同等[対等]者；同僚；

全て同じ役割をするノードたちの作りだすネットワーク
サーバとかクライアントとかの区別がない

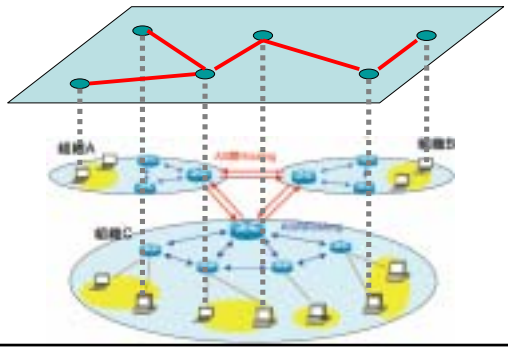
P2Pはアプリケーション層？

- 対等な関係で作るネットワークといえば、
  - Ad-hoc Networkでは、それを構成する全てのノードが平等な役割をになっていた
  - ルータが階層的に区切るネットワークは存在せず
  - すべてがフラットなネットワーク、すべてのノードがルータ的な役割りを果した
- しかし、世の中の大半の広域ネットワークはインターネット
  - よって、P2P的なネットワークはインターネットの上位層で形成されている

またの名をオーバレイネットワーク

- 下位層がどのようなネットワークで構成されていても、上位層にフラットなネットワークを作る
  - これが現代的なP2Pネットワーク
- 通信したい相手の識別子さえわかれば、そのホストと通信できる
  - これはIPアドレスを使えば、インターネットではできて当たり前
- ちょうど、下位層のネットワークの上に被せるように構成するネットワークであるために
  - オーバレイネットワークとも呼ばれる

オーバレイネットワークのイメージ



P2P的につながって何をするのか？

- 最初はファイル交換ソフトウェアから始まった
  - ぼくはこんなファイルを持っている
  - 私はこれだけ持っています
  - こっちにはこんなのがあるよ
  - うーん、それちょうだい
- 何がどこにあるのかを解決する手法が競われた
  - こんなコンテンツがどこどこのホストにある
- その後、次世代のファイルシステムとしての研究が始まり
  - Oceanstore, FARSITE, Ivy, Pangea
- Skypeが登場する
  - サーバレスで電話の機能をもつオーバレイネットワークをインターネット上に構築した
  - 数百万人が参加してもスケールする

P2Pネットワークに参加するとは？

- アプリケーション層で仲間のノードとのリンクを生成すること
- 誰が「仲間」であるかを誰が知っているか？
  - どこかに管理サーバがいて、そこにすべてを登録していくんだろう？
  - 何がどこにあるかわかったら、P2Pに接続する
  - これがもっとも旧タイプ(第1世代P2P)で、NapstarやWinMXなどが利用していた
  - 管理がしやすい、どんなデータが流れたかわかる
  - しかし、スケールしない
  - 参加者が増えたら爆発する
    - Single point of failureの存在

そもそも管理サーバってPeerじゃないよな！

- よし、サーバレスにしよう
  - 第2世代P2P、GnutellaとかWinny
- 自分の自分の近傍だけの知識をもっている
  - でも全部はつながっているいるのだから何とかなるだろう。
  - Unstructured P2Pと呼ばれた
- ローカルな情報を収集してそれを、互いに交換する

匿名性を獲得したネットワーク

- Freenetに代表されるネットワーク技術
  - 究極の民主主義を目指したともいわれる
- 日本ではWinnyなどが有名
- 自分が誰と話しているかわからない
  - もちろん話している直接の相手はわかるが、
  - 必ずしも通信のオリジンがそうとは限らない
- 一方では効率も重視される
  - BitTorrentやWinny
  - Aggressive replication技術

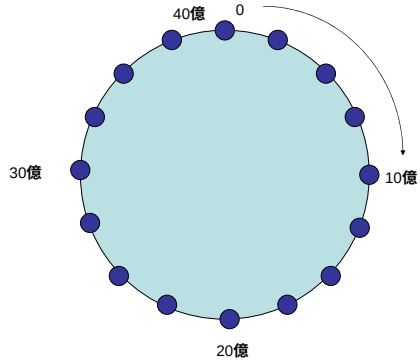
さらにstructuredなP2Pへ

- 第3世代と呼ばれるP2Pシステム
- どのコンテンツはどのホストに格納されるべきかをhash関数で管理
- しかし、このように大空間のhashはローカルには管理できないので、分散hashテーブル(DHT)というアルゴリズムが考案される
  - Chord, pastry, tapestry, soba

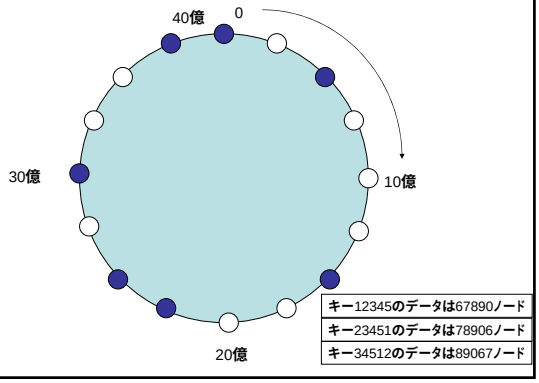
DHT:分散ハッシュテーブル

- コンテンツを表わすキーワードからハッシュ値を計算する関数を用意する
  - ハッシュ空間は数十億 (SHA-1の場合)
- そのキーワードを含むコンテンツを持つノードは、そのハッシュ値のノードIDを持つノードが知っている
- しかし、そんなに多くのノードは存在しないので、そのノードが存在しない場合にはその直前のIDのノードが知っている

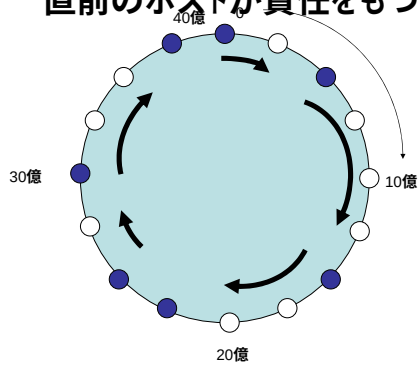
ハッシュ値を円形に並べてみると



でも実際に存在するホストはスカスカ

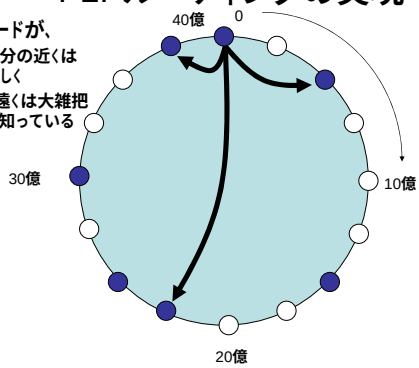


存在しないホストの役割は直前のホストが責任をもつ



後はどうやってそのホストに到達するか P2Pルーティングの実現

- 各ノードが、
  - 自分の近くは詳しく
  - 、遠くは大雑把に知っている



後の問題は

- 新しいノードが参加するときの処理
  - 各ノードの処理や担当配分を再分担
- ノードが離脱したときの処理
  - 離脱することが事前にわかっていればいいが
    - 関連ノードに処理を委託して抜ける
  - 突然、離脱したら
    - 例えば近隣ノードが多重化して保持しておく