

IPv6 ネットワークにおける エニーキャスト通信実現のための プロトコル設計と実装

土居 聡

大阪大学 大学院情報科学研究科
情報ネットワーク学専攻 博士前期課程

E-mail: s-doi@ist.osaka-u.ac.jp

2002/6/27

ITRC 研究会

1

発表の内容

- エニーキャストとは
- エニーキャスト通信の性質
 - 問題点の提示
- 研究の目的
- 新たな通信プロトコルの提案
 - 設計と実装
 - 実機による評価
- エニーキャストルーティング
 - サーバ負荷分散モデル

2002/6/27

ITRC 研究会

2

エニーキャストアドレス (1)

- 機能 (サービス) に対して割り当てられるアドレス
 - 複数のインターフェースの集合を識別
 - グループのうち「最適な」インターフェースに配送
 - 「最適さ」は経路制御機構によって決められる

IPv6 のアドレスタイプ

アドレス	アドレス設定の対象	通信形態	通信対象
ユニキャスト	ノード	1 対 1	1
マルチキャスト	グループ	1 対 多	多
エニーキャスト	機能 (サービス)	1 対 1	多 (その内の 1)

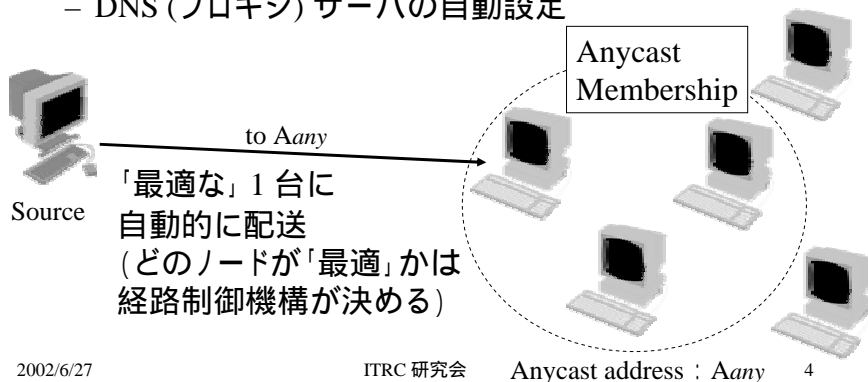
2002/6/27

ITRC 研究会

3

エニーキャストアドレス (2)

- 応用例
 - ミラーサーバでの利用
 - DNS (プロキシ) サーバの自動設定



2002/6/27

ITRC 研究会

Anycast address : Any

4

エニーキャスト通信の性質

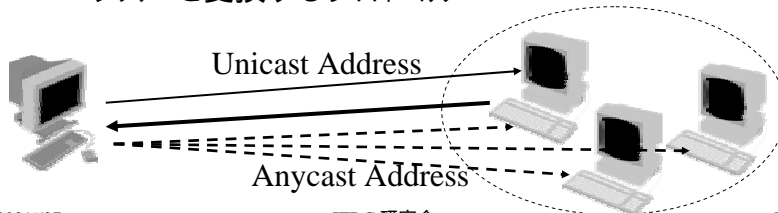
- 常に「最適な」ノードへとパケットを送信
 - 送信した複数のパケットが同じノードに届くとは限らない
 - 1パケットのみの送信には利用できる
 - 1つのノードとの接続を続けることはできない
- エニーキャスト通信への要求事項
 - 複数のパケットにより通信する場合
 - 最初の1パケットで最適なノードが選ばれた後は同じノードと継続して通信を行いたい

研究の目的

- 既存のアプリケーションを修正することなくエニーキャスト通信を実現する
- ネットワークの状況に応じた動的な端末設定を実現する

Anycast Address Resolving Protocol (AARP) の提案

- エニーキャストアドレスを最適なノードを調べるためだけに利用し、実際の通信はユニキャストアドレスに対して行う
- Anycast Address Resolving Protocol
 - エニーキャストアドレスを対応するユニキャストアドレスへと変換するプロトコル



2002/6/27

ITRC 研究会

7

AARP の実装方法

- エニーキャストアドレスを対応するユニキャストアドレスへ変換する方法
 - エニーキャストアドレスを割り当てられたノードからのパケットを受信する必要
 - 2つの方式
 - Client initiate プロープパケット方式
 - AARP をクライアント側で行う
 - Server initiate ビギーバック方式
 - AARP をサーバ側で行う
- アプリケーションを修正しないための方法
 - 中間ライブラリ (AARP ライブラリ) を作成し共有ライブラリを置き換える
 - 共有ライブラリのメカニズムを利用

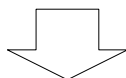
2002/6/27

ITRC 研究会

8

両方式の比較

	プローブパケット方式	ビジーバック方式
問題点	最初の一往復のプローブパケット分だけ多くのパケットを必要とする	アプリケーション層、トランスポート層のすべてのプロトコルを修正する必要がある
利点	実装が容易	トラフィック量はユニキャストと同じ



- Client initiate プローブパケット方式を採用
- プローブパケット: ICMPv6 Echo Request/Reply

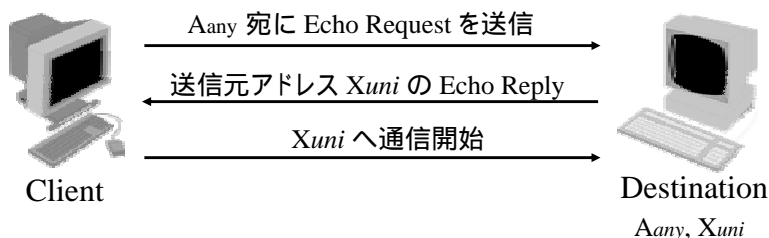
2002/6/27

ITRC 研究会

9

ICMPv6 Echo Request/Reply を用いたアドレス解決

- エニーキャストアドレス宛に Echo Request を送出し、端末からの Echo Reply を受信することでユニキャストアドレスを取得する

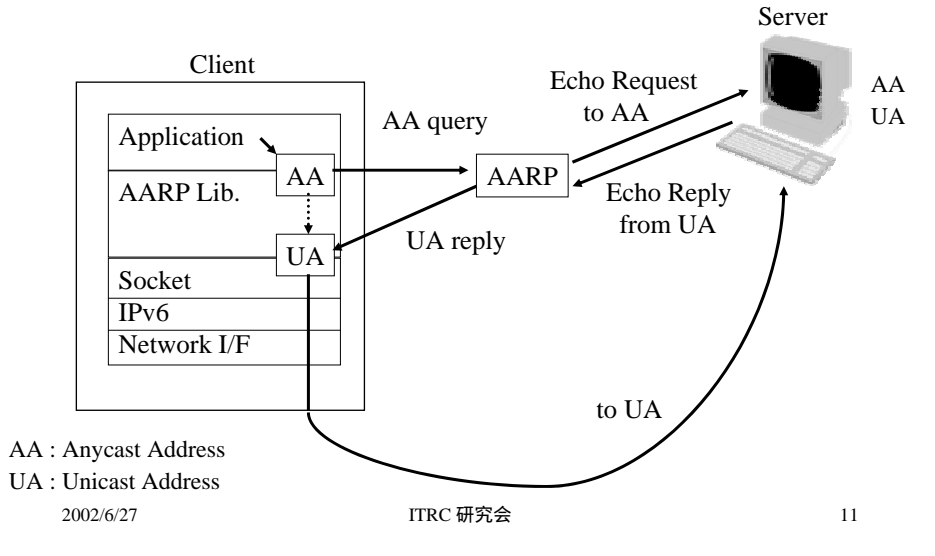


2002/6/27

ITRC 研究会

10

AARP ライブラリ



提案方式の評価 (1)

- FreeBSD 4.4 – Release 上で実装評価
- 環境変数 LD_PRELOAD に AARP ライブラリを設定
- TCP 通信の評価
 - telnet, ftp によるエニーキャストアドレスを割り当てたノードとの通信が実現できているかどうかの評価実験
- UDP 通信の評価
 - DNS サーバにエニーキャストアドレスを割り当て、/etc/resolv.conf にそのエニーキャストアドレスを設定し、ホスト名解決が行われているかどうかの評価実験
- オーバーヘッド: 一往復分の Echo Request/Reply

提案方式の評価 - telnet



```
> setenv LD_PRELOAD libarp.so  
> telnet Sany (Anycast)
```

2002/6/27

ITRC 研究会

13

エニーキャストルーティング

- 同じエニーキャストアドレスを割り当てられた複数のノードが同一セグメント上に存在するかどうかによって経路情報の更新方法が異なる
 - 同一セグメント上に存在する場合
 - Neighbor Discovery (ND)
 - 異なるセグメント上に存在する場合
 - RIPng, OSPFv3 等のルーティングプロトコル

2002/6/27

ITRC 研究会

14

Neighbor Discoveryによる サブネット内での端末選択

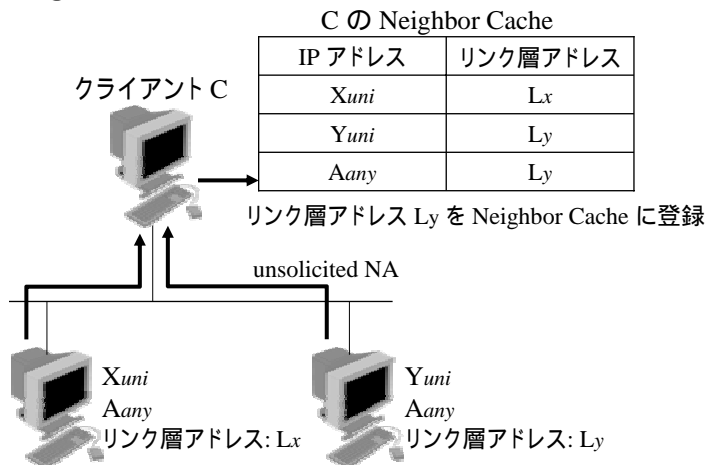
- 宛先 IP アドレスに対応するリンク層アドレスが必要
 - Neighbor Cache によって決まる
- Neighbor Cache
 - IP アドレスとリンク層アドレスを組にしたもの
 - Neighbor Discovery によって更新
 - キャッシュの保持時間が過ぎると破棄
 - 通常、Neighbor Solicitation (NS) に対する Neighbor Advertisement (NA) を受信したとき更新される
 - 近隣ノードの到達性判断に用いる

2002/6/27

ITRC 研究会

15

unsolicited NA を用いた Neighbor Cache 更新手法の提案



2002/6/27

ITRC 研究会

16

unsolicited NA を用いた サーバ負荷分散モデル

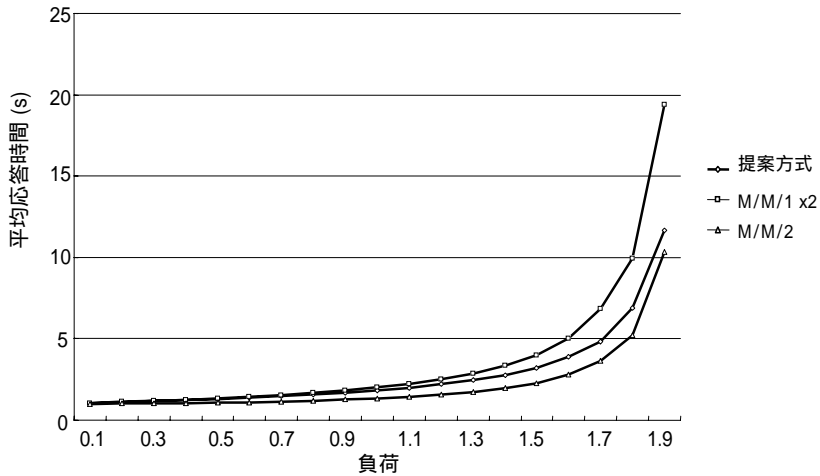
- 同一セグメント内に 2 台のエニーキャストサーバを配置
 - 各サーバが unsolicited NA を送出することで動的にルータの Neighbor Cache を更新
 - サーバは T 秒毎に unsolicited NA を送信
-
- エニーキャストアドレス宛のパケットはより負荷の小さなサーバに届けられる

シミュレーションによる 負荷分散モデルの性能評価

- 待ち行列を用いたシミュレーションによる評価
 - サービス要求は到着順 (FIFO) で処理
 - サービス要求はレート λ のポワソン到着
 - 各サーバにおけるサービス時間は平均 $1/\mu$ の指数分布に従う

⇒ 2つの $M/M/1$ 待ち行列によりモデル化
- 比較対象は $M/M/1 \times 2$ (DNS ラウンドロビン), $M/M/2$ (理想的) 待ち行列
- 負荷 ρ は CPU 利用率

シミュレーション結果



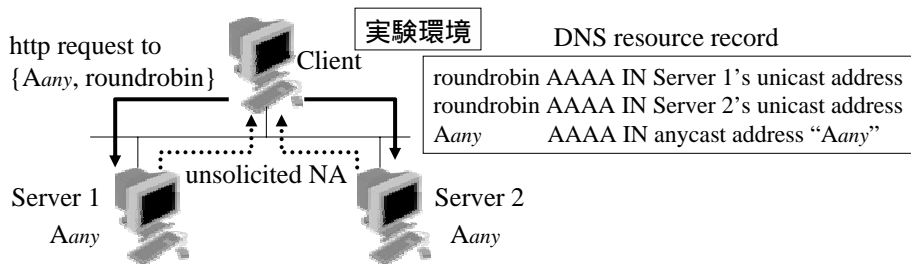
2002/6/27

ITRC 研究会

19

実機による負荷分散モデルの性能評価

- 2 台の Web サーバに提案方式を実装し、http リクエストの平均応答時間を計測
- 各サーバ上で CPU に負荷をかけるプロセスを実行
- 比較対象は DNS ラウンドロビン



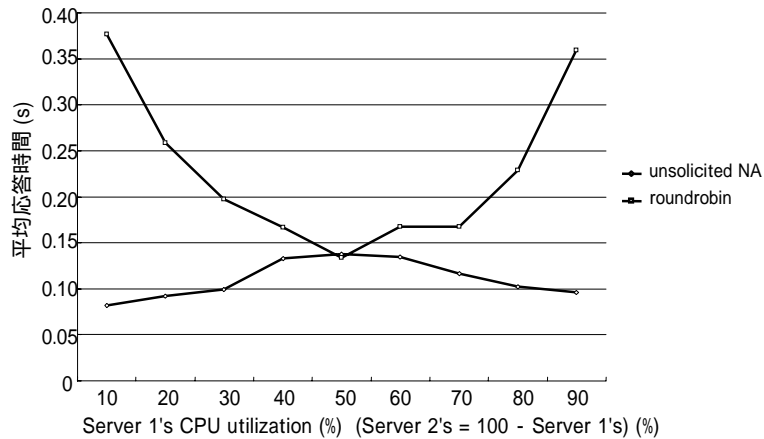
2002/6/27

ITRC 研究会

20

実機による評価結果 (1)

平均応答時間



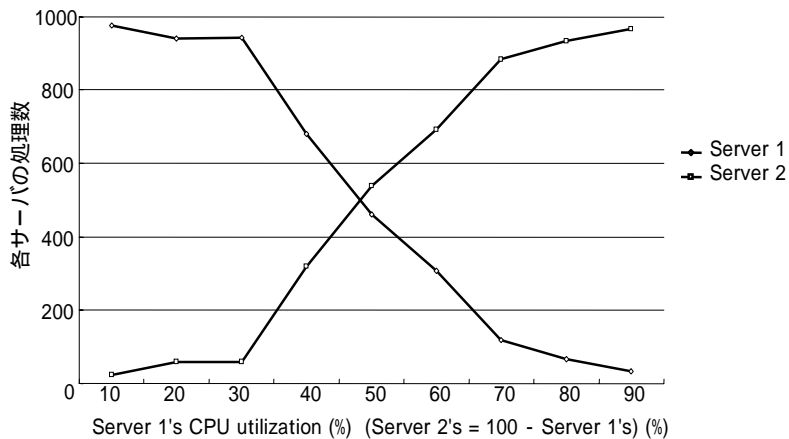
2002/6/27

ITRC 研究会

21

実機による評価結果 (2)

各サーバのリクエスト処理数



2002/6/27

ITRC 研究会

22

まとめと今後の課題

- **まとめ**
 - エニーキャストアドレスの応用例と利用時の問題点に関する考察
 - エニーキャスト通信を実現するためのプロトコル設計と実装および評価
 - サブネット内での端末選択方式の提案
- **今後の課題**
 - ICMPv6 Echo Request/Reply によるメッセージ量の把握
 - アドレス解決のキャッシングを行う場合のキャッシュの保持時間の検討
 - 同じエニーキャストアドレスを割り当てられた複数のノードが異なるセグメント上に存在する場合の経路制御機構の確立