

# 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

宮原研究室  
土居 聡

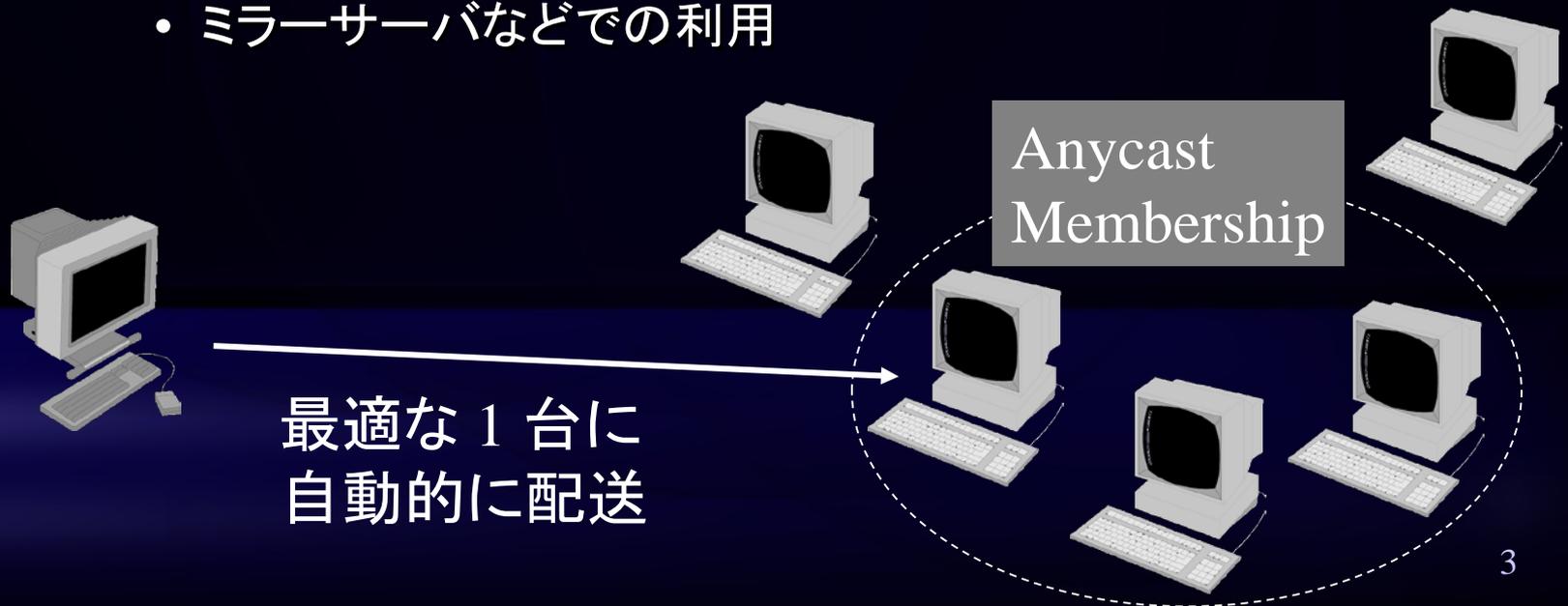
# IPv4 から IPv6 へ

- アドレス空間の拡大
  - 32 ビット → 128 ビット
- 新たな機能の追加
  - セキュリティ
  - リアルタイム通信
  - 移動端末のサポート

アドレスタイプ	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
<b>エニーキャスト</b>	機能 (サービス)	1 対 1	多 (その内の 1)

# エニーキャストアドレス

- 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
    - ミラーサーバなどでの利用



# 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



既存の枠組みを変更する必要

# 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

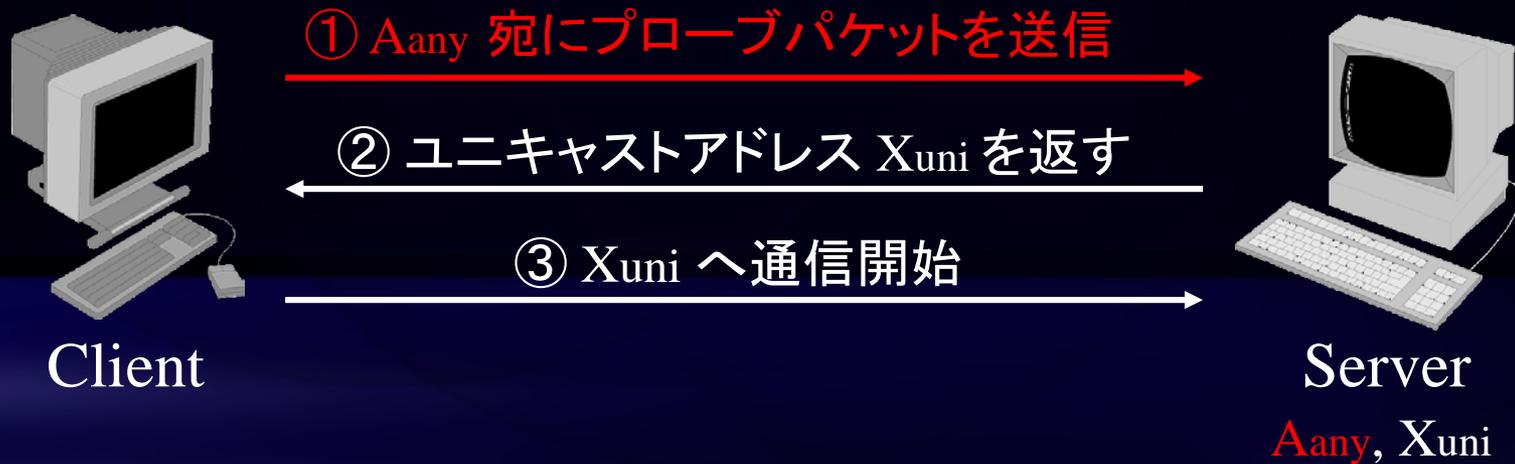


# AARPの実装方法

- 2つの方式
  - Client initiate プローブパケット方式
    - AARPをクライアント側で行う
  - Server initiate piggyback 方式
    - AARPをサーバ側で行う
- 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
  - 既存のアプリケーションプログラムを修正することなくアプリケーションの挙動を変えることができる

# Client initiate プローブパケット方式

- クライアントがエニーキャストアドレス宛のプローブパケットを送出し、端末からの応答を受信することでユニキャストアドレスを取得する



# Server initiate piggyback 方式

- サーバからの通信パケット上に、エニーキャストアドレスを付加 (piggyback) させ、エニーキャストアドレスとユニキャストアドレスを対応させる



# 両方式の比較

	プローブパケット方式	Piggyback 方式
問題点	プローブパケットによるトラヒックが発生し、ネットワーク資源を消費してしまう	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	トラヒック量はユニキャストと同じ



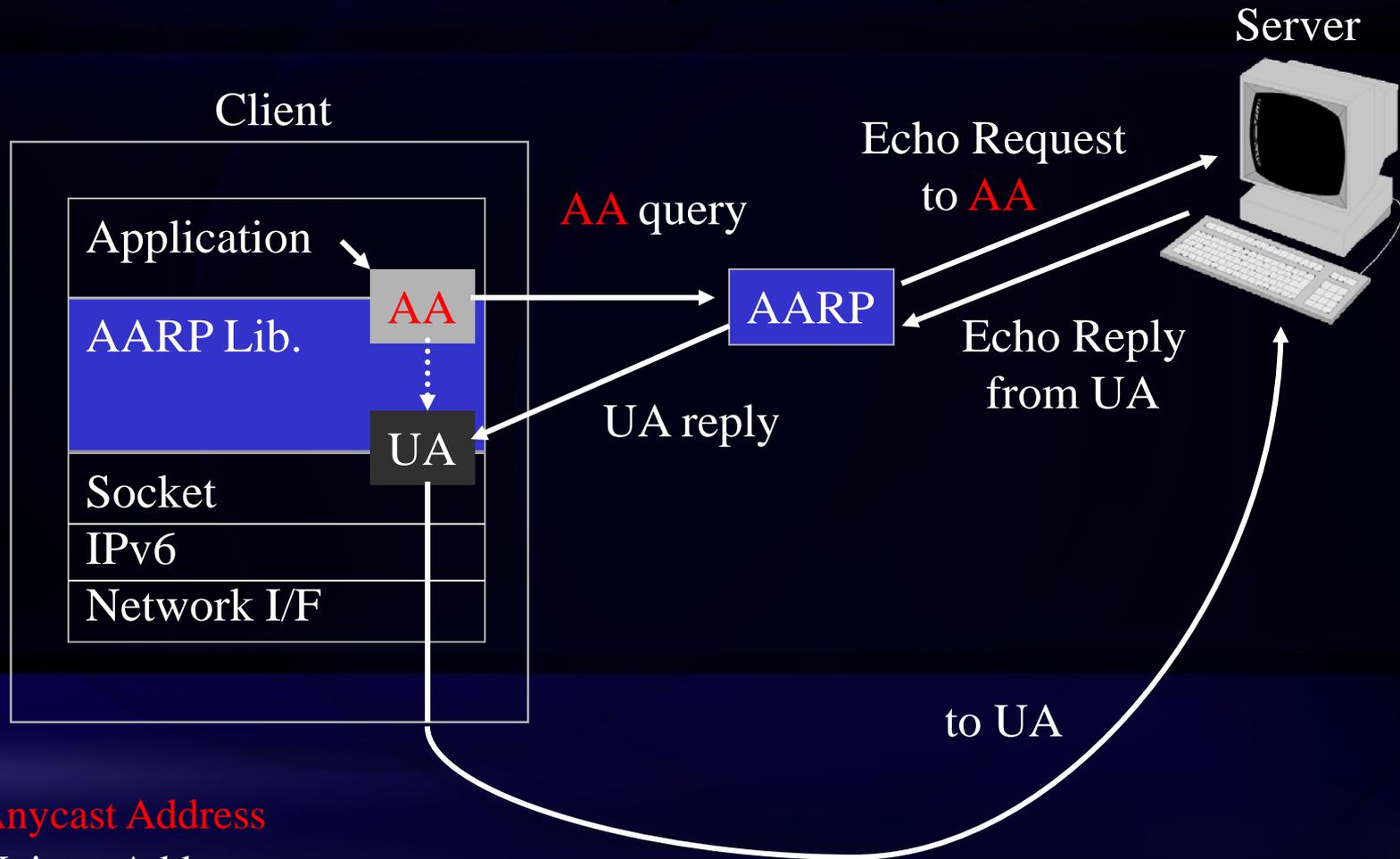
- Client initiate プローブパケット方式を採用
- プローブパケット: ICMPv6 Echo Request/Reply

# 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



# AARP ライブラリ



AA : Anycast Address

UA : Unicast Address

# 提案方式の評価

- FreeBSD 4.4 – Release 上で実装評価
- 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/resolve.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



エニーキャスト通信の実現性を確認

# まとめと今後の課題

- 本報告のまとめ
  - 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
- 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
  - エニーキャストアドレスのための新たな経路制御機構