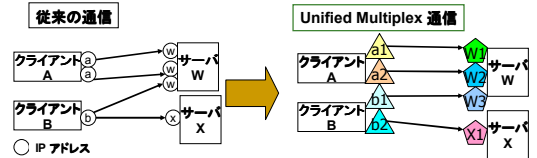


Unified Multiplex 通信アーキテクチャ におけるアドレス管理機能の設計

大阪大学 基礎工学部 情報科学科
村田研究室
西田 和生

研究の背景 1/2

- Unified Multiplex 通信アーキテクチャ
 - 毎回接続を行うたびにアドレスが変化 (IPv6 環境)
- サービス専用アドレスを実現
 - クライアントのロケーションプライバシーを保護
 - サーバのセキュリティ向上
 - アドレスを知ることの出来ない第三者はサーバに対して接続不可能



研究の背景 2/2

- 従来の通信と Unified Multiplex 通信では以下の項目で異なる

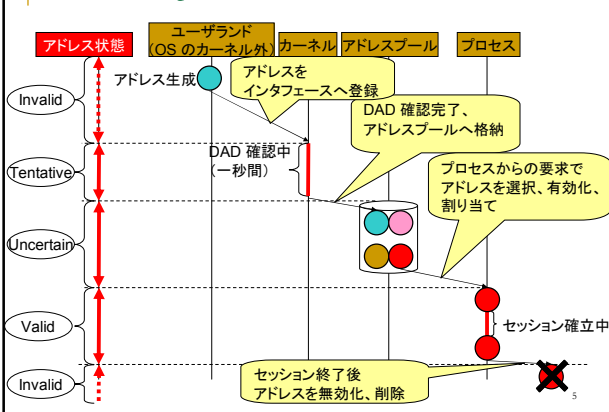
	従来の通信	Unified Multiplex 通信
アドレスの個数	1つのみ	多数設定利用
アドレスの使用目的	ノード識別子	サービス識別子
アドレスの有効期間	ノードが起動し終了するまで	セッション開始からセッション終了まで
アドレスの変化	固定で変化しない	更新され変動する

- ユーザによるアドレス設定は現実的ではない
 - Unified Multiplex 通信はアドレスが動的に変化するため

研究の目的

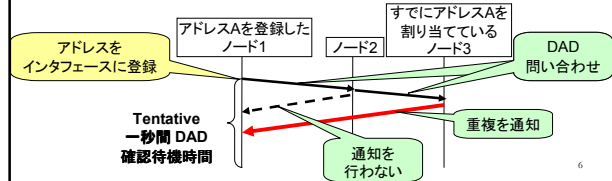
- サービス専用アドレスを実現する Unified Multiplex 通信のためのアドレス管理機能の設計
 - サービス専用アドレスを実現するための生成、削除までのアドレス状態遷移を分析
 - 各状態遷移を実現するための管理機能を設計
- サービス専用アドレスとは
 - セッション開始時にアドレスを割り当て
 - セッション終了時にアドレスを削除
 - 使用するアドレスは毎回異なる

Unified Multiplex 通信のアドレス状態遷移



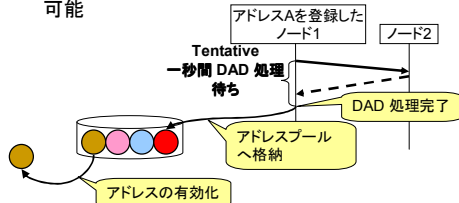
Invalid 状態から Tentative 状態への遷移

- アドレスのインタフェースへの登録
- アドレスの重複検出を行うことが必要
- 重複検出には DAD を使用
 - IPv6 の仕様で規定
 - アドレスが他のノードと重複していないことを確認
 - 検出待機時間に一秒割り当てられている



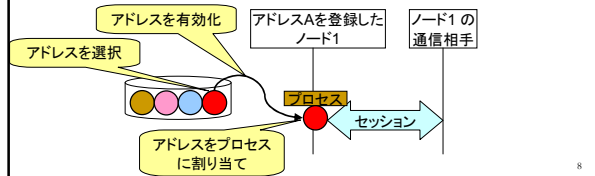
Tentative 状態から Uncertain 状態への遷移

- DADを完了後、アドレスプールへ格納が必要
- Uncertain 状態
 - 従来の IPv6 のアドレス状態遷移には無い状態
 - DAD が完了している状態
 - 有効状態である Valid 状態の前の状態
 - アドレスプール内に格納、有効化することで通信に使用可能



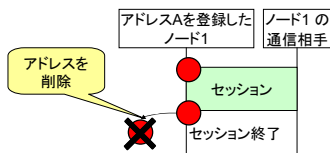
Uncertain 状態から Valid 状態への遷移

- 適切なアドレスを選択することが必要
 - 大量にアドレスが存在するため
- アドレスを有効化することが必要
 - アドレスを通信に使用するため
- プロセスに割り当てる必要がある
 - プロセスが通信を行うため



Valid 状態から Invalid 状態への遷移

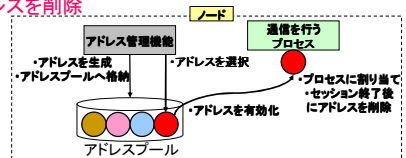
- アドレスをインタフェースから削除することが必要
 - アドレスを他の通信に使用させないため
 - セッション終了後に行う
- アドレスの無効化
 - アドレスを削除することで無効化



アドレス管理機能

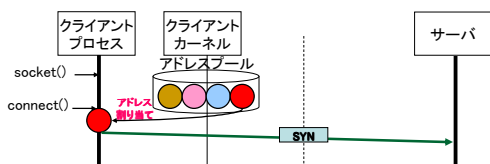
- アドレス状態遷移の分析を基にアドレス管理機能の設計

- アドレスを生成
 - アドレス値の生成方法、アドレスの重複検出
- アドレスをアドレスプールへ格納
- アドレスを選択
- アドレスを有効化
- アドレスをプロセスに割り当て
- アドレスを削除



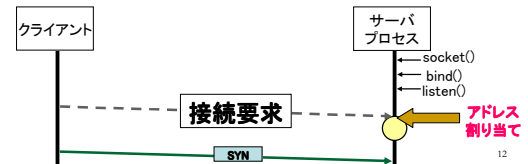
アドレス管理機能 ~ アドレスを割り当て ~

- クライアント
 - 送信元アドレスを選択する in6_selectsrc() 関数を変更
 - アドレスプール内からアドレスを選択、送信元アドレスとして割り当て
 - アドレスを割り当てる際、Uncertain 状態から Valid 状態へ遷移させるため、in6_flags の IN6_IFF_UNCERTAIN を削除する
- 上記の変更により、connect() の実行時にプロセスに対してアドレスを動的に割り当てるが可能



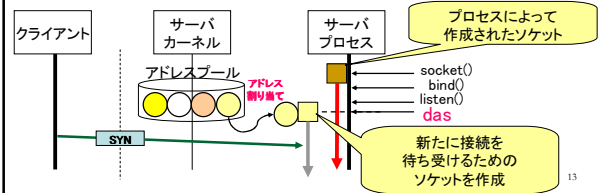
アドレス管理機能 ~ アドレスを割り当て ~

- サーバ
 - クライアントから接続要求があった際にアドレスを割り当てる必要がある
 - listen() した後にアドレスを割り当てる機能が必要になる
 - DAS (Delayed Address Setting) の導入
 - bind() でアドレスの設定を行わない
 - 新たに LISTEN ソケットに対してアドレスを割り当てる
 - listen() 後にアドレスを割り当てる
 - das コマンドの作成



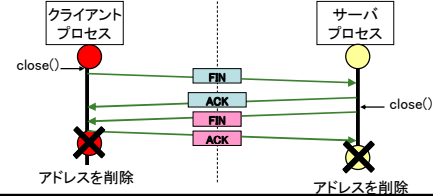
アドレス管理機能 ~ アドレスを割り当て ~

- das コマンド
 - 引数として pid、インタフェース、アドレスを指定
 - LISTEN ソケットを取得する fgetfirstsock() 関数を作成
 - 新たに待ち受け用ソケットを作成し、プロセスのソケットに割り当てる in6_umx_ioctl() 関数を作成
- 上記の変更により、サーバはクライアントからの接続要求があった場合のみ待ち受け可能になる



アドレス管理機能 ~ アドレスを削除 ~

- 正常に通信が終了した際
 - クライアント、サーバともに
 - close() が呼ばれ、セッション終了 (TCP)
 - アドレスを削除
- 正常に通信が終了しない際
 - セッション情報を管理している PCB の削除
 - アドレスを削除



まとめと今後の課題

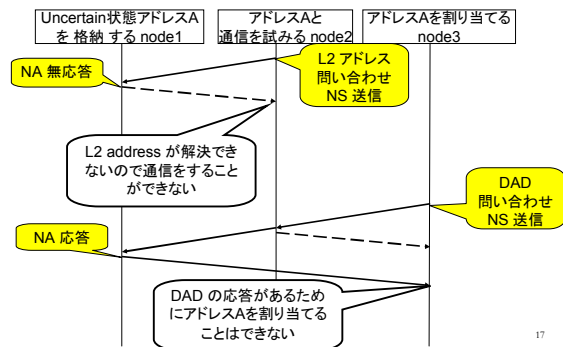
- サービス専用アドレスを実現する Unified Multiplex 通信アーキテクチャ
 - Unified Multiplex 通信のアドレス管理機能を設計
 - プロセスからの要求に動的に対応
 - ユーザが様々な手順を踏むことなく Unified Multiplex 通信を利用可能
 - アドレス管理機能は実装済み (Unified Multiplex 通信は FreeBSD 6.2 上で実装されている)
- 今後の課題
 - アドレス管理機能に、より高度な制御を導入
 - アドレス使用状況に応じたアドレス生成
 - アドレス生成に特別な仕組みを導入し、新しいサービスの提供

15

ご清聴ありがとうございました

16

Uncertain 状態の導入



DNSO

