動的ネットワーク設定プロトコルの改良と評価

冨永明宏[†,☆] 寺岡文男[††] 村井純[†††]

本稿では，ネットワーク単位のアドレス管理を実現する Dynamic Network Configuration Protocol（DNCP）を改良する．第1に，広域ネットワークでも運用できるようにスケーラビリティを改善する．そのためにネットワークをコア部分と末端部分の2つに分類し，各々で動作を変化させる．特にコア部分には，論理的な関係を基にグループを構成する階層的グループ化の概念を導入し，安定性を向上させた．第2に，IP アドレスの利用率を監視する Policing 機構を導入し，IP アドレスの効率的な割当を実現する．第3に，不正なホストからの要求を排除する認証機構を実現する．本稿では，これらの改良を含めて DNCP を実装し，評価実験を行った．実験では，実際に小規模なネットワークを用いてテストするとともに，ネットワークエミュレータを用いて大規模ネットワークを再現し，処理性能と IP アドレスの利用率を測定した．その結果，実用的な処理速度で利用率の高い IP アドレス管理を実現できることが分かった．最後に，インターネットへの応用可能性について考察した．

Improvement of Dynamic Network Configuration Protocol and Its Evaluation

Akihiro Tominaga,[†,☆] Fumio Teraoka[††] and Jun Murai[†††]

This paper improves Dynamic Network Configuration Protocol (DNCP) that manages IP addresses for networks. First, scalability is improved to enable DNCP operation in the wide area network. Networks are categorized into “Core” and “Leaf” parts, and new DNCP works differently in each part. Specially, “Hierarchical Grouping”, which forms groups based on the logical relation between servers, is introduced and improves stability of DNCP. Then, the policing mechanism that keeps the rate of used IP addresses within a range is introduced for efficient IP address management. Next, the authentication mechanism is added to exclude requests from invalid hosts. This paper implements improved DNCP and evaluate it. Several experiments to check functions of DNCP were done in a small network, and other experiments to measure performance were done in a large network with simulation. Results show that processing speed were enough while achieving efficient IP address management. Last part of this paper considered application of DNCP to the Internet.

1. はじめに

現在のインターネットでは，IP アドレスの割当や付け替えは主として手作業で実施している．そのため，ネットワークの構成，規模，接続コンピュータの激しい変化などに迅速に対応できず，IP アドレス利用率にも無駄が生じている．近年になって，Dynamic Host Configuration Protocol（DHCP）[1),2)]や IPv6 Auto Configuration[3)]のような IP アドレスの自動割当機構が開発された．しかし，これらは単体のホストが対象で，ネットワーク単位でのアドレス自動管理を実現するのは，筆者らが提案した Dynamic Network Configuration Protocol（DNCP）[4)]のみである．

文献 4) では，プロトタイプ実装により DNCP の基本動作を検証したが，本稿では，DNCP を実際のネットワークで運用するための改良をいくつか加える．第1に，文献 4) では組織内ネットワーク程度を管理対象としていたが，本稿では広域ネットワークでの運用を目指してスケーラビリティを向上させる．第2に，適切なシステム運用に不可欠な Policing 機構を追加し，IP アドレスの使用率を向上させる．第3に，IP アドレスの不正取得を排除するための認証機構を追加する．

† 慶應義塾大学大学院政策・メディア研究科
Graduate School of Media and Governance, Keio University
☆ 現在，ソニー株式会社インフォメーション&ネットワーク研究所インターネットシステムラボ
Presently with Internet Systems Laboratory, Information & Network Technologies Laboratories, Sony Corp.
†† 株式会社ソニーコンピュータサイエンス研究所
Sony Computer Science Laboratories, Inc.
††† 慶應義塾大学環境情報学部
Faculty of Environmental Information, Keio University

これらの改良を含む形で DNCP を再設計し，評価を行うのが本稿の目的である．また，DNCP を広域インターネットに適用する方法についても考察する．

2. DNCP の目的

2.1 DNCP の対象領域

近年のネットワークの普及にともない，ネットワーク構築に必要な IP アドレスの割当・設定作業が，管理者の負担となっている．また，経路情報の急増や IP アドレスの枯渇といった問題に対し，IP アドレスの割当効率を向上させつつ経路情報を統合して経路数を減少させる CIDR という技術が運用されている[5]．これにともない，CIDR を用いるのに必要なネットワークの IP アドレス付け替えを自動化する機構が求められている．

以上をふまえ，IP アドレスの自動管理機構を構築することが，DNCP の目的である．ただし，DNCP では IP アドレスの付け替えにともなう通信断絶の問題や各種アプリケーションの自動設定変更などの問題は扱わない．またホスト名と IP アドレスのマッピングの動的更新については，外部機構[6]に一任する．

2.2 DNCP のアドレス管理モデル

DNCP では，ネットワークに起こる変化の中で，アドレス管理に関係するものを以下の 4 通りに分類する：

- ネットワークが新たに付け加わる（**追加**）
- ネットワークが取り除かれる（**削除**）
- ネットワークが別の位置へ移動する（**移動**）
- ネットワークアドレスを付け替える（**付け替え**）

これらすべては，IP アドレス群の「割当」と「返却」の 2 つの操作で処理する．たとえば「ネットワークの追加」では，必要な IP アドレス数を見積もって割り当てる．「ネットワークの削除」では，使っていた IP アドレス群を返却する．インターネットサービスプロバイダ（ISP）の変更は「ネットワークの移動」の例で，この場合は新しい IP アドレス群の割当を受け，古い IP アドレス群を返却する．

「付け替え」のみ，ネットワーク接続が変化しない☆．たとえば，ネットワーク内のホストが増えたためさらに多くの IP アドレスが必要だが，隣接するアドレス空間は割当できない場合や，割当と返却を繰り返すうちに細分化されたアドレス空間を整理するときにアドレス付け替えを行う．付け替えでは古い IP アドレス群を回収し，新しい IP アドレス群を割り当てる．

☆「ネットワークの移動」もアドレス付け替えを引き起こすが，本稿ではネットワーク接続が変化しないものを特に区別する．

2.3 用語定義

本稿で使用する用語を以下のように定義する：

- **サブネット** ··· 物理リンク，それを共有する 1 台以上のルータ，複数のホストの 3 つから構成される．各サブネットには最低 1 台のルータが存在する．また，各ルータは 2 つ以上のサブネットに接続される．
- **アドレスプレフィクス** ··· IP アドレスのネットワーク部を示し，「先頭 IP アドレス/有意部分のビット数」の形で表現される．たとえば「192.168.0.0/16」とは，先頭の 16 ビットが共通の IP アドレスの集合（192.168.0.0〜192.168.255.255）を表す．
- **アドレスバルク** ··· アドレスプレフィクスで表される IP アドレスの集合のこと．分割したり，逆にあるアドレスバルクの部分集合がすべて揃っている場合には統合できる（例：192.168.0.0/16 ↔ 192.168.0.0/17, 192.168.0.128/18, 192.168.0.192/18）．アドレスバルクの集合を「アドレスプール」と呼ぶ．
- **ネットワークトポロジー** ··· ネットワークトポロジーとは，サブネット群，各サブネット上のホスト群，サブネット群をつなぐルータ群，物理リンクなどから構成される，総合的なネットワークの構造を表す．

3. スケーラビリティの改善

3.1 DNCP の基本動作

DNCP では，個々の IP アドレスの割当を受けるホストを DNCP クライアントと呼ぶ．DNCP サーバは，DNCP クライアントおよび他の DNCP サーバに，IP アドレスやアドレスバルクを割り当てる．

IP アドレスの設定前でも DNCP が動作するように，通常の IP 経路制御とは独立の通信機構を用いる．そのため，管理対象のサブネット群に含まれる全ルータは DNCP サーバの機能を持つ．さらに，様々な物理リンクで利用できるように，IP（UDP）上に構築する．

また，管理者の負担を軽減するため，中央の DNCP サーバに管理すべきアドレスプールを与えると，他の DNCP サーバにアドレスバルクが配分されるようにする．ただし全 DNCP サーバが，1 台の中央サーバから直接アドレスバルクを取得すると負荷が集中するうえに，中央サーバの故障がシステム全体に影響を及ぼす．これを避けるため，DNCP サーバ群は中央の DNCP サーバ（ルートサーバ）を根とする木構造を構築し，それに沿って階層的に IP アドレスを管理する．

アドレスバルクの管理は，「上位 DNCP サーバ（木構造の根に近い方が上位）から下位 DNCP サーバへのアドレスバルク割当」と「下位 DNCP サーバから上位 DNCP サーバへのアドレスバルク返却」の 2 操

作を基本とする．返却は，IP アドレスを付け替える場合や，余分な IP アドレスを回収する場合に用いる．割当要求を受けた DNCP サーバは，保有するアドレスバルクを適切な大きさに分割して割り当てる．アドレスバルクが足りなければ，さらに上位の DNCP サーバに対してアドレスバルクの割当を要求する．返却を要求された場合も，必要に応じて保有するアドレスバルクを分割あるいは統合する．

個々の IP アドレスの管理も，「DNCP サーバから DNCP クライアントに対する IP アドレス割当」と「DNCP クライアントから DNCP サーバに対する IP アドレス返却」で実現する．各 DNCP サーバは定期的にサブネット情報をアナウンスし，これを受信したクライアントが IP アドレスの割当を要求する．

3.2 スケーラビリティ改善の必要性

文献 4) の設計では，アドレス管理に用いる木構造はネットワークトポロジーに基づいて自動構築される．具体的には距離ベクトル型の経路情報交換プロトコル[7),8)]に似た手法により，DNCP ルートサーバからのホップ数に基づく Spanning Tree を生成する．

しかし，この手法には 2 つ問題がある．第 1 に，ネットワークトポロジーの変化が木構造の変化を招く．変化した地点から下位のサーバでは IP アドレス付け替えが発生するため，大規模ネットワークのルートサーバ付近でネットワークトポロジーが変化すると，影響が広範囲に及ぶ．第 2 に，従来の木構造構築方法ではネットワークが大規模になるにつれて階層構造が深くなる．DNCP では，将来の割当要求に備えて各サーバが未割当 IP アドレスを保有しているため，階層が深くなると未割当 IP アドレスが累積し，全体の IP アドレス利用率が低下する．

また，CIDR の効果を最大限に発揮するためには，できるだけ広範囲にわたって協調的に IP アドレスを管理しなければならない．一方，IP アドレス付け替えを容易にする DNCP をさらに規模の大きなネットワークで運用するためには，これら 2 つの問題を解決し，DNCP のスケーラビリティを改善する必要がある．

3.3 木構造構築機能の再設計

インターネットの経路制御は，スケーラビリティや安定性を確保するために，ドメイン内経路制御とドメイン間経路制御の 2 層に分離されている．OSPF[9)] などのドメイン内経路制御では，ネットワークトポロジーに応じて自動的に経路が決定されるのに対し，BGP4[10)] などのドメイン間経路制御では，管理者が半ば静的に経路を設定することも多い．

同様に DNCP でも，アドレス管理機構を 2 レベルに分ける．ドメイン間経路制御に相当する部分（コア部分）では，ネットワーク構成があまり変化しないと仮定し，ネットワーク管理の階層に基づいてサーバの木構造を構築する．ただし，論理的な管理階層を自動決定するのは困難なため，あらかじめ管理者が木構造を指定する．コア部分では IP アドレスや IP 経路制御を用いて通信を行う．これにともない，コア部分では DNCP クライアントへの IP アドレス割当は行わず，DNCP サーバへのアドレスバルク割当のみを行う．

さらにドメイン間経路制御がドメインを単位として扱うように，コア部分でも複数の DNCP サーバからグループを構成する．各グループには，グループを代表してアドレスバルクの取得や返却を行う「代表サーバ」を 1 つ定義する．複数のグループやサーバをまとめて，より大きなグループを構成することも可能で，このような再帰的なグループの構成法を「階層的グループ化」と呼ぶ．これによって，コア部分では木構造の安定性を増し，スケーラビリティを向上させる．

一方，ドメイン内経路制御に相当する部分（末端部分）では，文献 4) 同様に，ネットワークトポロジーから木構造を自動構築する．また，末端部分では個々のホストに対する IP アドレス設定を実現するため，IP アドレスや IP 経路制御に依存しないようにする．

階層的グループ化の例を**図 1** に示す．末端部分 A，B は各組織のネットワークに相当し，各々がグループを構成する．末端部分内の最上位 DNCP サーバは，対外リンクを持ち，かつ代表サーバでなければならない．また末端部分 A，B を除いた残りがコア部分に相当し，点線は管理者の定義した論理的上下関係を表す．さらに末端部分 A，B と DNCP サーバ S1 が，より大きなグループを構成し，S1 がその代表サーバとなる．さらに S3 は「S1 を代表サーバとするグループ」と「S2 を代表サーバとするグループ」から構成される，より大きなグループの代表サーバとなる．

ある 1 台のルータが故障する確率を p とする．従来

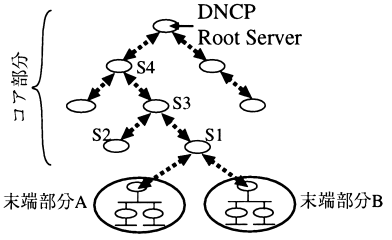

図 1 階層的グループ化の例
Fig. 1 An example of Hierarchical Grouping.

のようにネットワークトポロジーから階層構造を構築する場合，仮にルートサーバ〜S1 間が 10 ホップあると，いずれかが故障する確率は $10p$ となる．したがってホップ数が増えるほど，階層構造が変化する確率も高くなることが直観的に分かる．ところがコア部分で階層的グループ化を用いると，各サーバ間（RS〜S4 間，S4〜S3 間，S3〜S1 間など）に IP レベルの到達性がある限り階層構造は変化しない．したがって，大規模なネットワークでも階層構造が不安定にならないため，結果としてスケーラビリティが向上する．

このように，アドレス管理機構を 2 レベルに分けることで，全体としてのスケーラビリティを確保しつつ，末端部分ではサブネット群の IP アドレスを自動設定できる．さらに，末端部分の代表サーバが一括してアドレスバルクを取得するため，末端部分内では連続する IP アドレスを利用することになる．これは末端部分の出口（= 代表サーバの持つ対外リンク）において，CIDR による経路情報の集約が容易になるというメリットをもたらす．

4. Policing 機構の設計

各 DNCP サーバは，サブネットごとに必要な IP アドレスの量を見積もって割当を要求する．しかし見積りが不的確だったり，DNCP クライアントの減少などにより IP アドレスが余ることもある．各 DNCP サーバによる余剰 IP アドレスの自主的な返却を期待するだけでは適切なアドレス管理は成立しないため，Policing 機構を追加する．以下では Policing とアドレス管理アルゴリズムを同時に示す．また簡単な動作例も示す．

現在の設計では，10 秒ごとに DNCP サーバは次の処理を繰り返す．第 1 に，サブネットごとに IP アドレスの過不足をチェックする（**サブネット毎 Policing**）．第 2 に，自己のアドレスプールをチェックし，IP アドレスが不足していないかチェックする（**自己 Policing**）．第 3 に，各下位サーバのアドレスプールに余剰がないかチェックする（**下位サーバ毎 Policing**）．

4.1 サブネット毎 Policing

サブネット毎 Policing では，各サブネットごとに「サブネット内 IP アドレス割当率」を計算する．これは，サブネットに割り当てたアドレスバルクの IP アドレス数と，実際に DNCP クライアントが利用中の IP アドレス数の比率を表し，以下の式から求められる．ネットワークアドレスとブロードキャストアドレスの分，分母で 2 を減算している．

$$\frac{\text{アドレスを割当済の DNCP クライアント数}}{\text{(アドレスバルクの IP アドレス数}-2\text{)}}$$

あるサブネットに対し，割当済みのアドレスバルクも割当要求中のアドレスバルクも存在しない場合には，保有するアドレスプールから割当を行う．もし，アドレスプールに割当可能なアドレスバルクがなければ，上位サーバにアドレスバルクの追加割当を要求する．

サブネット内 IP アドレス割当率が上限を超えていて，かつ割当要求中のアドレスバルクが存在しない（あるいは十分に大きくない）場合は，IP アドレスが不足しつつあるものとして，より大きなアドレスバルクに付け替える．この際も，保有するアドレスプールで付け替えを実施できなければ，上位サーバにアドレスバルクの追加割当を要求する．逆に，サブネット内 IP アドレス割当率が下限を切っている場合には，IP アドレスが余りつつあるものとし，より小さなアドレスバルクへ付け替える．なお，今回は上限値を 0.90，下限値を 0.40 とした．

4.2 自己 Policing

自己 Policing では，自らの「サーバ内アドレスバルク割当率」を計算する．この値は，保有するアドレスプールに含まれる IP アドレスの合計数と，下位サーバやサブネットに割り当てたアドレスバルク群に含まれる IP アドレスの合計数の比率を表すもので，以下の式から求められる：

$$\frac{\text{割当済アドレスバルク群の IP アドレス総数}}{\text{アドレスプールの IP アドレス総数}}$$

この値が上限を超えている（IP アドレスが不足しつつある）場合は上位サーバにアドレスバルクの追加割当を要求する．今回は上限値を 0.90 とした．

4.3 下位サーバ毎 Policing

下位サーバ毎 Policing では，定期的に下位 DNCP サーバのサーバ内アドレスバルク割当率を問い合わせ，応答値に基づいて Policing を行う．具体的には下位サーバのサーバ内アドレスバルク割当率が下限を切った場合，適正範囲に収まるような大きさのアドレスバルクの返却を要求する．今回は下限値を 0.50 とした．

DNCP では，下位サーバの返すアドレスバルク割当率が正しいか検証できない．そのため，正しく認証された下位サーバが報告する値はつねに信頼するものとする．これは，IP アドレスの手動管理における Policing 機構でも，報告者を信頼するほかはないのと等しい．

4.4 DNCP の動作例

図 2 に，DNCP によるアドレス割当の例を示す．図中左上が管理対象のネットワークトポロジーで，S1〜

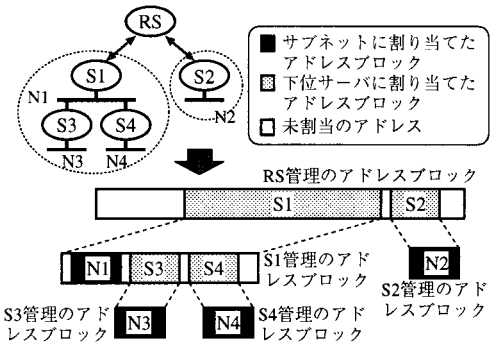

図 2 DNCP によるアドレス割当の例
Fig. 2 An example of address allocation with DNCP.

S4 は DNCP サーバ（兼ルータ）を表す．RS は DNCP ルートサーバを，N1〜N4 はサブネットを，長方形は各 DNCP サーバの管理するアドレスバルクを表す．

管理者の設定により，S1，S3，S4 のグループと，S2 単独のグループの 2 つが存在する．S1 と S2 は各グループの代表サーバであり，ルートサーバ RS との上下関係が静的に定義されている（図の両矢印）．S1 が代表サーバのグループは，内部で木構造構築機能を用いて上下関係を決定する．この例では，S3 と S4 が代表サーバ S1 の下位に決定される．その後，上下関係に基づいて「RS から S1 と S2 へ」「S1 からサブネット N1 と下位サーバ S3，S4 へ」「S3 からサブネット N3 へ」などの割当が行われる．

たとえば，S1 は定期的に N1 のサブネット内 IP アドレス割当率を計算し，過不足が発生している場合にはアドレスの付け替えを行う（サブネット毎 Policing）．次に自分のサーバ内アドレスバルク割当率をチェックし，上限を超えている場合には，アドレスバルクの追加割当を要求する（自己 Policing）．最後に下位サーバ毎 Policing を下位サーバ S3 と S4 に対して実施する．問合せの結果，サーバ内アドレスバルク割当率が低ければ返却要求を送信する．

5. 認証機構の設計

5.1 認証のモデル

不正な DNCP サーバや DNCP クライアントによる IP アドレスの占有を防ぐため，受信した DNCP メッセージは認証チェックする必要がある．認証は，DNCP サーバ対 DNCP サーバや DNCP サーバ対 DNCP クライアントなどのホスト単位で行う．DNCP サーバや DNCP クライアントは，認証に成功した DNCP メッセージだけを処理する．

全ホストで常時利用可能かつ完全に一意な識別子を用意するのは難しい．そのため必要に応じて識別子を選択する．今回は末端部分で MAC アドレス，コア部分で IP アドレスを認証用識別子として用いる．

5.2 認証方式と鍵管理問題

一般的には，「共有秘密鍵方式」「公開鍵方式」の 2 つの認証方式がある．共有秘密鍵方式の強度はハッシュ関数と共有秘密鍵の選び方で決まり，Keyed MD5 [11),12)] や HMAC MD5 [13)] などの規格が有名である．公開鍵方式は，秘密鍵を配布する必要がないので一般に安全性が高い．公開鍵方式の強度も暗号化に用いる関数によって異なり，有名な規格としては RSA [14),15)] がある．

末端部分では，通信相手となる DNCP サーバが一定でない．公開鍵方式は通信相手ごとに鍵が異なるため，特に DNCP を広域ネットワークに適用する場合に鍵管理方法が問題となる．また，DNCP では IP による通信が不可能なこともあり，問題はさらに複雑になる．

共有秘密鍵方式では，鍵管理の手間は多少軽減される．しかし，広域ネットワーク全体で 1 つの共有秘密鍵を用いると，共有秘密鍵が流出してしまう危険性が高い．そのため，ほぼ上下関係が静的なコア部分では各々の上位/下位サーバの組ごとに個別の共有秘密鍵を用い，各末端部分では個別の共有秘密鍵を用いるという運用上の解決方法が考えられる．

鍵配布問題は複雑なため，今回は認証の枠組みを用意して基本動作を検証するにとどめる．安全かつスケーラブルな鍵管理機構の実現は今後の課題とした．したがって，現在は各 DNCP サーバ・DNCP クライアントの管理者が手動で鍵情報を設定しなければならない．

任意の認証方法を採用できるように，DNCP メッセージには認証方法を指定するフィールドを設ける．ただし，HMAC MD5 をすべての DNCP サーバが実装しなければならない標準の認証方法とする．DNCP クライアントも HMAC MD5 を実装するのが望ましいが，「各ユーザの手間を煩わして各 DNCP クライアントに鍵を設定するよりも，完全な Plug & Play を望む」場合などは認証なしでもよい．

5.3 認証アルゴリズムの詳細

HMAC MD5 による認証の詳細を示す．すべての DNCP メッセージには 16 オクテットの MD フィールドが存在する．メッセージの送信側は，MD フィールドを 0 に初期化し，通信相手に応じた共有秘密鍵を送信メッセージの前後に付加する．これを文献 13) に定義されているハッシュ関数に入力し，得られた Message Digest を MD フィールドにセットして送信する．

DNCP メッセージの受信側は，同様にして Message

Digest を計算する．計算結果が受信した Message Digest と一致すれば，その DNCP メッセージの内容と送信者の認証は成功したものとする．認証に失敗した場合には，それ以上の処理は行わない．

6. 評　　価

BSD/OS 3.1 に実装し，小規模な実ネットワークおよび仮想環境で評価実験を行った．

6.1 小規模ネットワークにおける評価実験

2.2 節で示した 4 通りの変化についてテストした．さらに認証機構と Policing 機構も動作を確認した．

6.1.1 ネットワークの追加

図 3 にネットワーク追加実験用のネットワークの構成と実験結果の概略を示す．この実験では，初期状態で DNCP ルートサーバ A に，アドレスバルク 192.168.0.0/22 を与えた．DNCP サーバ B が起動すると，1/4 に分割された 192.168.0.0/24 が割り当てられた．さらに DNCP サーバ C と DNCP クライアントからなるサブネットを追加すると，このサブネットには 192.168.0.0/26 が割り当てられた．またサーバ C には 192.168.0.1，DNCP クライアントには 192.168.0.2 の IP アドレスが設定され，相互に通信できた．

このように，DNCP がネットワークの追加に対応することを確認した．またスケーラビリティを考慮した設計のとおり，サーバ C の割当要求がサーバ A ではなくサーバ B で処理されることも分かった．

次にアドレスバルクの要求送信から割当までの時間を，上位/下位サーバともに K6 200 MHz，メモリ 128 MB の計算機で 20 回計測した結果，平均 10.133 msec かかった．上位サーバとの RTT の実測値（ping 100 回）は平均 0.386 msec だったことから，平均処理時間は $10.133 - 0.386 = 9.747$ msec と分かる．よって，少なくともすぐ上位のサーバにアドレスが十分にあれば，実用的な時間で処理されるといえる．

6.1.2 ネットワークの削除

ネットワークが削除される際は，事前に上位サーバに通知がある場合とない場合がある．前者の場合，上位サーバがアドレスバルクの返却を要求する．後者の場合，手動でアドレスバルクの返却処理を実行させる必要がある．さもなければ「一時的なネットワーク分断ではない」と判断できるだけの時間が過ぎてから，アドレスバルクを回収したものとして自動処理する．

最初に，図 3 の DNCP サーバ C のプロセスを終了させることで「事前通知なし」の削除をテストした．今回はアドレスバルクを自動回収する機能は実装しなかったため，サーバ B に「サーバ C が消失した」と表示されることのみを確認した．

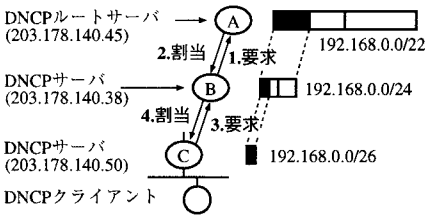

図 3 小規模ネットワークでの「追加」実験
Fig. 3 Network addition test in the small network.

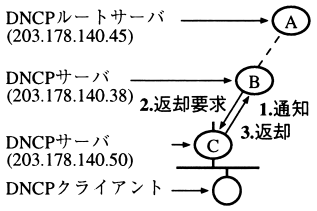

図 4 小規模ネットワークでの「削除」実験
Fig. 4 Network deletion test in the small network.

「事前通知あり」の削除テストでは，DNCP サーバ B からの通知に応じて上位サーバ A が返却要求を送信し，返却が行われることを確認した（**図 4**）．これらの結果から，DNCP は「ネットワークの削除」という変化にも対応できることが確認された．

また，1 つの返却要求の処理時間を先ほどと同様の環境・方法で計測した結果，平均 2.241 msec だった．少なくとも返却を要求したアドレスバルクがすぐ下位のサーバに未割当で残っている場合は，やはり実用的な時間で処理されるといえる．なお，割当要求と返却要求の処理時間の差は，前者は上位サーバが適切な大きさのアドレスバルクを得るための分割処理を含むために生じると考えられる．

6.1.3 ネットワークの移動

ネットワークの移動も，事前に上位サーバに通知がある場合と，突然移動する場合がある．事前に通知が行われる場合は「削除」と「追加」を組み合わせるだけなので，実験を省略する．移動後に通知がある場合については，図 3 の DNCP サーバ C を DNCP ルートサーバ A の下に移動させる実験をした．その結果，C からの通知に応じて，元の上位サーバ B がアドレスを回収することを確認した．また C が新しい上位サーバ A からアドレスを取得することを確認した（**図 5**）．

6.1.4 アドレスの付け替え

「アドレスの付け替え」は，割当済みのアドレスバルクに対して返却要求を送ることで実現する．実験は図 3 に示した「ネットワークの追加」実験に続けて行った．その結果，**図 6** に示すようにアドレスバル

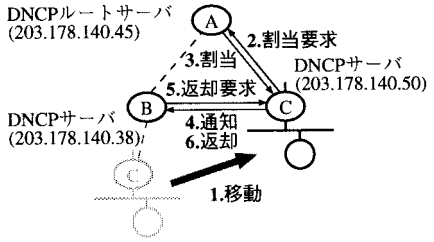

図 5 小規模ネットワークでの「移動」実験
Fig. 5 Network migration test in the small network.

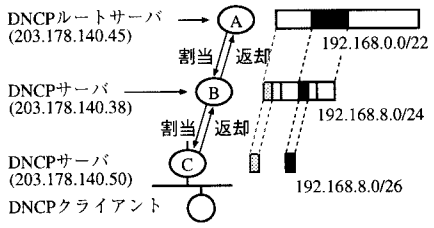

図 6 小規模ネットワークでの「付け替え」実験
Fig. 6 Renumbering test in the small network.

ク 192.168.0.0/24 が DNCP サーバ B から回収され，代わりに 192.168.8.0/24 が割り当てられた．これに先立ち，192.168.0.0/24 の一部である 192.168.0.0/26 も 192.168.8.0/26 に付け替えられた．

6.1.5 認証と Policing について

以上の実験は，すべて認証機構を組み込んだ状態で実施した．末端部分では MAC アドレス，コア部分では IP アドレスを識別子として用い，認証方式は標準の HMAC MD5 を実装した．たとえば，ネットワーク追加の実験の際に，故意に DNCP サーバ B だけ誤った共有秘密鍵を設定したところ，DNCP ルートサーバ A でエラーメッセージが表示されることや，アドレスバルクの割当が受けられないことを確認した．

Policing 機構については，DNCP サーバ C を修正し，見掛け上の DNCP クライアント数がランダムに増減するようにした．その結果，イーサネット側インタフェースで IP アドレス割当率が上限を超えたり下限を切ることによって，IP アドレスの付け替えが行われることが観測できた．それにともなって，アドレスバルク利用率も変化し，DNCP サーバ B からアドレス返却の要求が送信されることも確認した．実験により，単純に現在の DNCP クライアント数を基に Policing を行うと，境界値付近でクライアント数が増減を繰り返した場合にアドレス付け替えが頻発するということが明らかになった．この問題を回避するためには，たとえばクライアント数を 1 分ごとに記録し，過去 20 分間の平均クライアント数（あるいは最大クライアント数）を用いて IP アドレス割当率を計算すればよい．

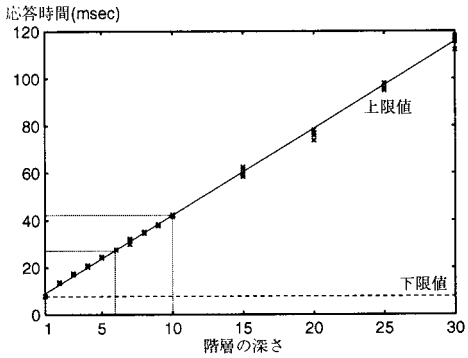

図 7 階層の深さと割当処理時間
Fig. 7 Hierarchy depth and time for the allocation.

実際に同様の変更を加えたところ，アドレス付け替えの頻度が下がることが観測できた．

6.2 エミュレータを用いた評価実験

ここでは，アドレスバルクの割当や返却の処理時間を定量的に評価することに主眼をおいた．実際の大規模ネットワークで性能評価実験を行うことは困難なため，ネットワークエミュレータによって仮想ネットワークを構築した．また途中で木構造が変化しないように，すべての実験にはコア部分だけを用いた．

実験では，パケットロスのない状態をエミュレートした．エミュレータも含む全プロセスは，K6 200 MHz，メモリ 128 MB，BSD/OS 3.1 の計算機で実行した．

6.2.1 アドレス割当に関する測定結果

初めにサーバの階層の深さと，割当要求が送信されてから実際に割り当てられるまでの時間を測定した．階層の深さは 1〜10 までは 1 段ずつ，以後は 30 まで 5 段ずつ増加させた（深さ 0 は DNCP ルートサーバしか存在しない状態なので扱わない）．各々について 5 回ずつ測定した結果を**図 7** にグラフ化する．

上限値（割当までに最も処理時間がかかるケース）を測定するため，中間の DNCP サーバが前もって余分なアドレスバルクを取得しないように実装を修正した．これにより，最下位の DNCP サーバの送信した割当要求は，必ず DNCP ルートサーバで処理される．

グラフから明らかなように，処理時間の上限値は階層の深さに正比例する．逆に最善の場合には，割当要求は 1 つ上位のサーバで処理されるため，処理時間は深さ 1 の場合に等しい（図 7「下限値」）．

次に，割当処理の際にサーバが送信するメッセージ数を評価した．プロトコルの仕様によれば，サーバの送信するメッセージ数の最小値は，階層の深さに関係なく 2 となる．これは 1 つ上位のサーバに必要なアドレスバルクがある場合に対応する．また，割

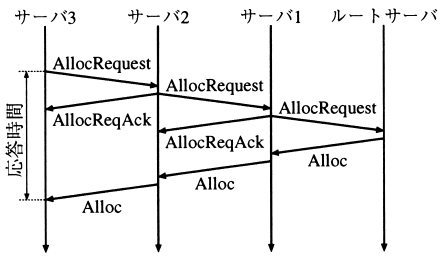

図 8 サーバ間のメッセージ交換の様子（階層の深さ 3）
Fig. 8 Message exchange among servers (depth 3).

当要求が DNCP ルートサーバで処理される場合にメッセージ数が最大となる．このとき，階層の深さを N とすると，サーバが送信するメッセージの総数は「$1+2(N-1)+N=3N-1$」という式で求められる．

先頭の「1」は最下位サーバが送信する返却要求を表す．次の「$2(N-1)$」は中間の $N-1$ 台のサーバが下位へ返す割当要求の Ack と，上位へ送信する返却要求を表す．次の「N」は最上位から最下位まで割当応答が N 回送信されることを示す．

実際にサーバ間で交換されるメッセージ数を計測し，計算式どおりの結果を得た．例として，階層の深さが 3 のときにサーバ間で行われる通信の様子を**図 8** に示す．計算式から分かるように，交換されるメッセージ数の上限値も階層の深さに正比例する．ただし，あるサーバとすぐ上位サーバの間で交換されるメッセージ数は，(再送が起きなければ) つねに 3 以下となることが図 8 からも読み取れる．

6.2.2 アドレス返却に関する測定結果

同じ方法・環境を用いて，返却要求の処理時間を測定した結果を**図 9** に示す．実験のため，最下位の DNCP サーバの保有するアドレスバルクを DNCP ルートサーバが返却要求するように実装を修正した．すべての返却要求は必ず最下位の DNCP サーバまで中継されて処理されるため，応答時間の上限値は階層の深さに正比例する．サーバ間の通信の様子も図 8 と同様で，メッセージ総数の上限値も同じ計算式で得られる．アドレス割当処理と同様に，メッセージ総数の最善値は 2 であり，応答時間の最善値は深さ 1 の場合と等しい．

6.2.3 アドレス利用率の比較

実際に人手で管理しているネットワークをエミュレータで再現し，DNCP で管理した場合と IP アドレスの利用率を比較する実験を行った．**図 10** にネットワークのトポロジーを，**表 1** に実験結果を示す．

本来は S2 をルートサーバにした方が効率的だが，故意に条件を少し悪くするため，S5 をルートサーバにした．表の 3，4 列目はサブネット内 IP アドレス

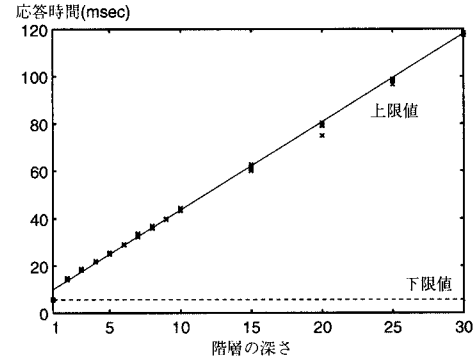

図 9 階層の深さと返却処理時間
Fig. 9 Hierarchy depth and time for release.

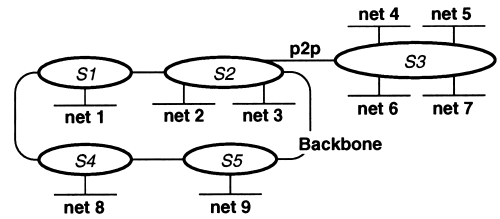

図 10 参照ネットワークのトポロジー
Fig. 10 Topology of the reference network.

表 1 サブネット内 IP アドレス割当率の比較
Table 1 Allocation rate of IP addresses in each subnet.

ネット名	ホスト数	手動	DNCP	比較
BB	15	50.00%	50.00%	
p2p	2	100.00%	33.33%	×
net1	18	60.00%	60.00%	
net2	8	57.14%	57.14%	
net3	1	7.14%	16.67%	○
net4	11	36.66%	78.57%	○
net5	19	30.65%	63.33%	○
net6	21	70.00%	70.00%	
net7	6	20.00%	42.86%	○
net8	10	33.33%	71.43%	○
net9	27	90.00%	43.55%	×

割当率を示す．実験の結果，2 つのサブネットを除いてすべて割当効率が向上した．割当率が低下した理由は，実装した管理アルゴリズムが「割当率が高すぎて DNCP クライアントの増加に即応できない」と判断して故意に余裕を確保したためである．割当率が 40%を下回っているサブネットもあるが，これは割り当てるアドレスバルクの最小の大きさを実装では制限したためである．

また，各サーバのアドレスバルク割当率も設計どおり 50〜90%の範囲に収まった．さらに手動管理では合計 302 個の IP アドレスが消費されているのに対し，DNCP では 250 個しか消費していない．つまり全体としても IP アドレスの利用率が向上した．

7. 考　　察

7.1 インターネットへの応用

現在は一部を除いて手動で IP アドレスが管理されている．Internet Assigned Number Authority（IANA）と呼ばれる組織が，IP アドレス管理の最高権限を持つ．IANA 単独では世界中を管理できないため，IP アドレス空間をいくつかのアドレスプールに分割し，その管理を下位の「地域 IR☆」に委任する．各地域 IR も委任されたアドレスプールを分割し，下位の「ローカル IR」に管理を委任する．さらにローカル IR は各国内の ISP に ··· と委任が繰り返され，最後には各ホストに IP アドレスが割り当てられる．このように IP アドレスの管理は，階層的な権限の委任により実現されている．現状の階層構造を**図 11** に示す．

この階層構造は DNCP と親和性が高い．図 11 の「IANA〜ISP」部は,「ISP〜最下層」部に比べてネットワークの構成や上下関係が安定している．したがって前者をコア部分，後者を末端部分として，DNCP を適用すればよい．こうすることで，末端のサブネット群は 0 から自動設定できるし，システム全体ではスケーラビリティと安定性を高く保てる．さらに ISP が，下位組織群に近接したアドレスバルクを割り当てることで，経路情報の集約に必要な条件が自然に満たされるようになる．ほかにたとえばローカル IR から各 ISP への割当なども「地理的に近いネットワークどうしは，ネットワークトポロジー的にも近い可能性が高い」ため，経路情報集約にも都合のよいことが多い．

7.2 処理時間について

サーバ間の RTT（Round Trip Time）は実際のネットワークの状況やサーバの配置によって大きく異なるので，実験では純粋な処理時間だけを測定することにした．しかし，実際には各サーバ間の RTT 分だけ応答時間が増えることが図 8 からも分かる．

図 7 や図 9 などに着目すると，階層の深さに比例して処理時間が増えるので，DNCP の性能が劣るように思われる．しかし，これは実際には処理時間の上限値を示しているにすぎない．たとえば，1 つ上位あるいは下位のサーバで要求が処理される可能性もある．その場合の処理時間は，深さ 1 の場合と等しくなる．すなわち深さ 1 の場合の処理時間が，下限値となる．

さらに現状のインターネットのコア部分では，図 11 に示したような深さ 5 程度の階層でアドレスが管理されている．末端部分でも，通常はバックボーン上のルータから 1〜3 ホップ程度ですべてのルータに到達できることが多い．したがって，現状ではアドレス管理の階層は深さ 6〜10 程度にしかならない．

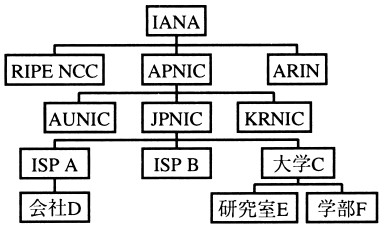

図 11　IP アドレス管理階層の現状
Fig. 11　Current hierarchy of the address management.

これらの現状を前出の図 7 に反映すると，応答時間の上限値は 27〜42 msec 程度にしかならないということが分かる．さらに，階層が 1 つ増えるごとに 3〜4 msec 程度しか上限値は上昇しないことも読み取れる．実際にはサーバ間の RTT を加算しなければならないが，それでもアドレス割当要求に対する応答時間は十分に実用的な範囲内に収まっている．

割当の場合は事前に各サーバに分配しておく余分なアドレスバルクの量を増やすほど，近い階層のサーバで処理される可能性が高くなる（ただし，IP アドレスの利用率は低下する）．しかし，返却の場合は何階層下の DNCP サーバが，返却対象のアドレスバルクを保持しているかは予測できない．さらに，返却対象のアドレスバルクが実際に使用されている場合，アドレス付け替えに要する時間も考慮しなければならない．

したがって，IP アドレスが緊急に必要になった場合には，アドレスを下位サーバから回収するよりも，上位サーバから割り当ててもらう方を優先するべきである．一方，返却処理は比較的長い時間をかけて下位サーバから余剰分を回収する方が適している．

8. おわりに

本稿では，ネットワーク単位のアドレス管理を実現する Dynamic Network Configuration Protocol を改良し，実装と評価を行った．第 1 に，広域ネットワークに適用するためにスケーラビリティを改善した．具体的にはネットワークをコア部分と末端部分に分類し，論理的関係を基に木構造を構築する階層的グループ化を導入した．第 2 に，適切な管理に不可欠な Policing 機構を実現した．実験の結果，アドレスバルクの割当率が指定範囲に収まること，手動管理より利用率が向上することを確認した．第 3 に，不正なホストからの要求を排除するのに必要な認証機構を設計・実装し，

☆ Internet Registry の略．IR は IP アドレスを割り当て，管理者の氏名・連絡先などをデータベースに登録（Register）する．

実験で動作を検証した．そのほかにも文献 4) で検証できなかったネットワーク単位のアドレス付け替えや，実ネットワークにおける総合試験を実施した．また IP アドレスの割当や返却の処理時間を測定し，実際のネットワークでは実用的な処理時間を達成できることを示した．最後に，広域インターネットへ DNCP を適用する可能性について考察した．今後は，広域インターネットでも利用可能な認証機構や鍵管理機構を実現したり，実際のネットワークで DNCP を運用して問題点を洗い出すなどの作業を行う必要がある．

謝辞 本稿執筆を暖かく見守ってくれた村井研究室と WIDE プロジェクトの諸氏，妻の祥子に感謝します．

参 考 文 献

1) Droms, R.: Dynamic Host Configuration Protocol, RFC 2131 (1997).
2) Tominaga, A., Osamu, N., Teraoka, F. and Murai, J.: Problems and Solutions of DHCP, *Proc. INET '95*, Vol.1, pp.481–490, ISOC (1995).
3) Thomson, S. and Narten, T.: IPv6 Stateless Address Autoconfiguration, RFC 1971 (1996).
4) 冨永明宏，寺岡文男，村井 純：階層的手法を用いた動的ネットワーク設定機構，コンピュータソフトウェア，Vol.16, No.1, pp.1–11 (1999).
5) Rekhter, Y. and Li, T.: An Architecture for IP Address Allocation with CIDR, RFC 1518 (1993).
6) Vixie, P., Thomson, S., Rekhter, Y. and Bound, J.: Dynamic Updates in the Domain Name System (DNS UPDATE), RFC 2136 (1997).
7) Huitema, C.: *Routing in the Internet*, Prentice Hall (1995).
8) Perlman, R.: *Interconnections: Bridges and Routers*, Addison Wesley (1992).
9) Moy, J.: OSPF Version 2, RFC 1247 (1994).
10) Rekhter, Y. and Li, T.: A Border Gateway Protocol 4 (BGP-4), RFC 1771 (1995).
11) Kaliski, B. and Robshaw, M.: Message authentication with MD5, *CryptoBytes* (*RSA Labs Technical Newsletter*), Vol.1, No.1 (1995).
12) Metzger, P. and Simpson, W.: IP Authentication using Keyed MD5, RFC 1828 (1995).
13) Krawczyk, H., Bellare, M. and Canetti, R.: HMAC: Keyed-Hashing for Message Authentication, RFC 2104 (1997).
14) Rivest, R.L., Shamir, A. and Adleman, L.: A method for obtaining digital signatures and public-key cryptosystems, *Comm. ACM*, Vol.21, No.2, pp.120–126 (1978).
15) Kaliski, B.: PKCS #1: RSA Encryption Version 1.5, RFC 2313 (1998).

(平成 11 年 5 月 14 日受付)
(平成 11 年 12 月 2 日採録)

冨永 明宏

1972 年生．1994 年慶應義塾大学環境情報学部卒業．1996 年同大学大学院政策・メディア研究科修士課程修了．1999 年同博士課程所定単位取得退学．同年ソニー（株）入社．コンピュータネットワーク，ネットワークの自動設定に興味を持つ．日本ソフトウェア科学会，ACM 各会員．

寺岡 文男（正会員）

1959 年生．1984 年慶應義塾大学大学院工学研究科電気工学専攻修士課程修了．同年キヤノン（株）入社．1988 年（株）ソニーコンピュータサイエンス研究所入社．工学博士．1991 年日本ソフトウェア科学会高橋奨励賞受賞．1993 年元岡記念賞受賞．コンピュータネットワーク，オペレーティングシステム，分散システム等の研究に従事．特に移動透過性を提供するプロトコル VIP（Virtual IP）の開発を通して IETF の Mobile IP 分科会の活動に貢献．著書に「ワイヤレス LAN アーキテクチャ」（共著，共立出版）．監訳に「詳解 Mobile IP」（共監訳，プレンティスホール出版）．ACM，IEEE，Internet Society，日本ソフトウェア科学会各会員．

村井 純（正会員）

1955 年生．1979 年慶應義塾大学工学部卒業．1981 年同大学院同学部修士課程修了．1984 年同大学院同学部博士課程修了．工学博士．1984 年東京工業大学総合情報処理センター助手．1987 年東京大学大型計算機センター助手．1988 年 WIDE プロジェクトを設立し，今日までその代表として指導にあたる．慶應義塾大学 SFC 研究所所長．社団法人日本ネットワークインフォメーションセンター理事長，インターネットソサエティ（ISOC）理事．ICANN 暫定ボード．