コンピュータネットワークにおける正しい構成発見のための情報認証方式

村山優子†

様々な環境下において，興味の対象となる物の存在とそれらの位置について知りたいという要求が発生する．本論文では，この構成情報取得要求を構成発見と呼び，コンピュータネットワークの代表的なものであるインターネットの環境下における問題点と解決法を探る．構成発見は，現在の環境構成についての情報を得ることで達成できる．この20年ほどの間にインターネットはめざましい規模増大を遂げてきた．このような大規模環境についての情報管理は動的学習や階層的管理域分割などにより解決されてきた．どちらの手法においても情報と現状の不一致や情報の不正確さの問題が発生する．この2つの観点から構成発見のための正確な情報の必要性を考察し，解決法の1つとして，構成情報の証明書を使用した情報認証を提案する．その応用例としてインターネットを構成する各サブネットでのアドレス証明書の使用によるアクセス制御を試みる．

Information Certification for Correct Configuration Detection in Computer Networks

YUKO MURAYAMA†

In computer networks, one needs to know which objects exist and where they are located. We call the acquisition of this knowledge Configuration Detection. We are interested in hosts and routers on an internetwork. As internetworking has become popular, the growth in the number of network objects has been substantial and dynamic. In such an environment, the maintenance of knowledge of the current configuration of the network has become a practical problem. The scale problem has been solved either by facilitating dynamic learning or by dividing the management domain into subdomains of a manageable size. In either solution, we face the problems of inconsistency and invalidity. We look into the serious consequences from these two problems, and propose a sloution, the use of a certificate for the authorisation of network addresses.

1. はじめに

情報化社会では，社会システムの機能が様々な情報に依存する．コンピュータネットワークにおいても同様である．そのようなシステムを情報依存システムと呼ぶことにする．情報依存システムでは，その動作機能は，使用される情報の正しさに左右される．コンピュータネットワークでは，その動作はネットワーク上のノードのアドレスなどの構成情報に依存する．本論文では構成情報を取得する行為を構成発見と呼び，正しい構成情報の必要性をセキュリティの立場から説き，情報認証方式を提案する．

ここ20年ほどの間のインターネットに代表されるコンピュータネットワークの発展における最も顕著な点はその拡大度である．インターネットが地球規模の通信ネットワークを成すまでに至った背景には，初期のネットワークの階層構造についての研究[1]に基づいた開放型アーキテクチャ[2]およびパケット交換技術[3]から生まれたパケット交換を行う網間ネットワークすなわちインターネットの形成技術の普及がある．ここ数年では，分散情報システムWWW（World-Wide Web）[4]の出現により，インターネットがさらに拡大した．この結果として，ネットワークは期待された以上の大きさとなり，限られた範囲でのネットワーク環境のためにしか設計されていない既存の管理システムでは対処できない様々な問題が浮上してきた．その1つに，現在のネットワーク環境内に存在するホストやルータ☆といったネットワークオブジェクトとそれら

† 広島市立大学情報科学部情報工学科
Department of Computer Engineering, Faculty of Information Sciences, Hiroshima City University

☆ ホストやルータはインターネットにおけるノードの名称で，ホストは終端システムでルータは中間システムである．

の位置をどのように知るかという問題がある．本論文では，これを構成発見と呼び，インターネットにおける問題点を探り，それらの問題の解決法として構成情報の認証を提案し，インターネッの構成要素であるサブネット内における応用を考察する．

以下，2章では構成発見の問題点を論じ，3章ではそれらの問題から派生する安全性などの問題点を示す．4章では3章にあげた問題点の解決法としての情報の認証を提案し，サブネットでのアドレス変換プロトコルでの証明書の応用を試みる．

2. 構成発見の定義とその問題点

インターネットの古典とされるShoch[5]の言葉を用いると，コンピュータネットワークの動作には次のような知識が必要である；

どのようなオブジェクトが存在し，
それらがどこに位置し，
それらにどのようにして到達できるか．

これらの知識に必要な情報が構成情報である．ネットワーク管理システムでは，管理オブジェクトの存在とそれらの位置，また，それらが管理装置からどのように監視および制御することができるかという知識が必要である．電子メイル・システムにおいては，メッセージの送信時に，受信者とそのアドレスを知ることが必要である．経路制御では，ルータやホストは使用可能な他のルータの存在とそれらを通してどの宛先ホストまで到達可能かなどの情報が必要である．このようなネットワーク動作に必要な構成情報を得る行為を構成発見と定義する．

本論文では，特に，管理域内のネットワーク上のホストやルータについての構成発見について考える．ネットワーク構成情報は，ネットワーク管理システムと情報システムの2種類のシステムにより管理される．したがって，どちらかのシステムに問い合わせることでネットワークの構成発見を行うことができる．ネットワーク管理システムは，ネットワーク管理作業のためにネットワークの構成情報を保持し，情報システムは，ネットワーク動作のための情報を提供する．情報システムには，ネームサーバ[6]やディレクトリ[7]などの他に，経路情報交換システムや[8]，アドレス変換システム[9]，MACレベルのブリッジシステム[10]などがある．

80年代から90年代に至る間のインターネットの拡大にともなうホストやルータの数の増加については，ネットワーク管理システムや情報システムは階層的な分離型（decentralised）管理方式[11]あるいは動的学習により対処してきた．

階層的分離型や動的学習で解決されたかに見える情報管理には，2つの問題が潜在していた．すなわち，情報が現状に則していないことから起こる一貫性の問題（*inconsistency*）と，情報自体が初めから間違って登録されてしまう不正確さの問題（*invalidity*）である．一貫性の問題は報告や登録なしにノードの追加や除去がなされた場合や逆に登録だけされたがノードの設定がなされていない場合に起こる．不正確さの問題は，ノードが間違って設定されてしまう場合にも起こる．これらの問題は，登録された情報についての検証がないことに端を発している．一貫性の問題と不正確さの問題は，安全性に対する脅威など，様々な波及効果をネットワーク環境に及ぼす．

安全性については応用層やネットワーク層では，それぞれ異なる要求が存在する．一貫性の問題と不正確さの問題から生ずる応用層の安全性の問題には他のノードシステムやサーバへの認可されていないアクセス，なりすまし，認可されていない情報の読み取りなどがある．これらの問題はそれぞれのアプリケーションでの解決が最も有効な手段であろう．本論文では，特にネットワーク層における不正確さの問題に着目する．次章では，これらの問題からの波及効果の考察を行う．

3. ネットワーク層における一貫性の問題と不正確さの問題の波及効果

3.1 誤った下位層アドレスによる問題

ネットワーク情報の不正確さは，ネットワーク環境に様々な問題を引き起こす．本節では，誤った下位層アドレスの設定により，その上のネットワークの動作に異常をきたす問題を取り上げ，これを考察する．次に下位層アドレス情報の不正確さから生じるネットワーク層の安全性の面での脅威を考察する．

これはイーサネット上の1つのホストのMACアドレスが放送アドレスに誤って設定されたときに起こる網の嵐（network storm あるいは broadcast storm）[10]と呼ばれる現象が起こることに着目し，発展させた問題である．網の嵐とは，パケットがネットワーク上に過剰に溢れ，ホストやルータのインタフェースにおいて輻輳が起こることである．イーサネット上にインターネット・プロトコル（IP）がおかれた環境を仮定して以下にこの現象を解説する．

図1のように，同じイーサネット上のホストAとルータBが同じMACアドレスに設定され，しかも，ホストAが転送機能を持つように設定されてしまったとする．通常ホストはルータと異なり転送機能はない

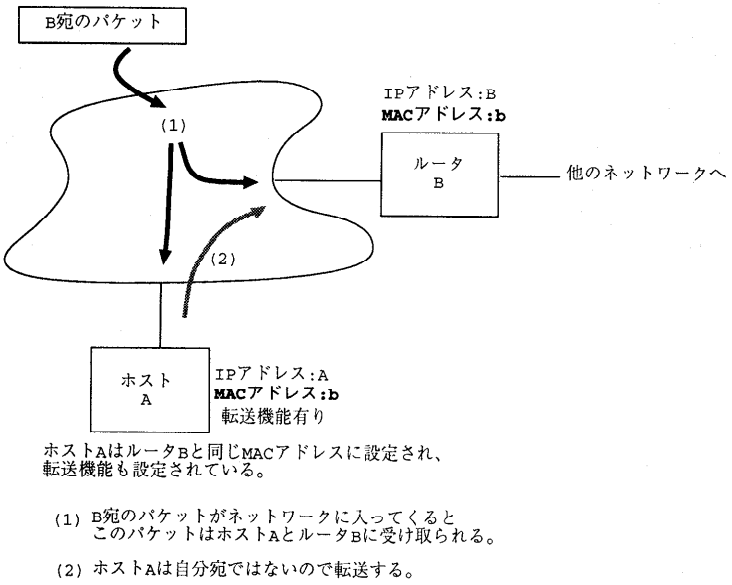


図1 トラフィックの倍増の様子
Fig. 1 Generating doubled traffic.

が，これらの設定は簡単なシステム構成情報の修正ミスで起こりうる．ただし，ホストAとルータBのIPアドレスは異なる．あるノードがルータB宛のパケットを送り出すと，このパケットは，同じMACアドレスに設定されたAとBで受け取られてしまう．Aは転送機能が設定してあるので，自分宛でない場合，受け取ったパケットを転送する．すなわち，Aにおいて余分なパケットが生成され，ルータB宛に送られることになる．発信ノードが生成したパケットが，サブネットを越えた外部あてのものの場合，余分なパケットの波及効果は外部へも伝わってしまう．このように，ルータB宛のトラフィックが倍増する．

また，逆にホストA宛のトラフィックは必ずルータBで受け取られ，転送されるので，このトラフィックも倍増する．このとき，ホストAがサーバである場合，倍増トラフィックによる輻輳の可能性も十分ある．

さらに，1つのサブネット上の3台以上のルータが同じMACアドレスに設定されてしまった場合には，網の嵐が起こる．3台のルータを同じMACアドレスに設定してしまったとする．この中の1台のルータが正規ルータである宛先に送られたパケットは，他の2台のルータでも受け取られ，再び正規ルータへと転送される．その際，再送パケットは，それぞれ，他の2台のルータで再び受け取られ，再送される．このように，1つのパケットは他の2台のルータで再生され続ける．各パケットにはTTL（Time To Live）と呼ばれる寿命が設定され，転送できる回数が制限されているので，この寿命パラメータがゼロになるまで，再生され続ける．TTLは指定できる寿命の最大値は255で，分散ファイル・システムなど，最大値を設定しているアプリケーションは少なくない．したがって，再生パケットは寿命に到達して廃棄される前に，すでにサブネット内を飽和状態にし，網の嵐という現象を引き起こす可能性は十分にある．

上記の状態を一般化すると，サブネット上でn個のルータ機能のあるノードが，同じMACアドレスに設定された場合，その1つに向けて送り出されたパケットは，まず，$n-1$個のルータで再生される．その後は，$n-2$個のルータで再生され続ける．寿命（TTL）の値がtに設定されていたとすると，そのサブネット上に再生されるパケットの総数は：

$$S = (n-1)+(n-1)(n-2)+\ldots+(n-1)(n-2)^{t-2}$$
$$= (n-1)\sum_{i=1}^{t-1}(n-2)^{i-1}$$

となる．これらは，余分なトラフィックであるが，正規ルータはこれらの再生されたパケットをすべてサブネットの外へと送り続ける．したがって，これはインターネット全体へのトラフィック増加につながる．

この状況は，あるホストのMACアドレスを放送アドレスに設定することにより，簡単に実現できてしまう．放送アドレス宛のパケットはサブネット上のすべ

てのルータで受け取られるからである．この現象は，この特定な環境だけに起こるわけではなく，IP のようなパケット交換方式が放送型の下位層ネットワーク上で使われるときに起こる普遍的な問題として考えることができる．

これは，Manber [12]が，「連鎖反応」(chain reaction) と呼んだ現象である．「連鎖反応」とは，コンピュータやネットワークプロトコルの設計上の盲点やミスがネットワーク上に波及し，最終的にネットワーク全体の崩壊を招くことである．

網の嵐状態を解除するには，誤った設定をされたホストやルータだけでなく，そのサブネット上の正常なルータをすべて停止しなければ，事態が収拾できないので，定常的な運用が求められるネットワークでは重大な問題となる．

網の嵐は起きないが，つねにトラフィックが倍増される場合，再生されるパケット群が正規パケット群であるため，いったん，サブネットの外に出た場合，その制御はだれにもできないところに重大性がある．一般にパケット交換ネットワークのルータでは転送したパケットについての情報はルータ内に残らない “stateless” と呼ばれる動作状態なので，ルータでの余分なパケット群の制御は不可能である．サブネット外へ出た余分なトラフィックはそれらが余分なものであるかどうかの判断は宛先ノードまで行かないとできない．したがって，これらの制御はサブネット内で行うべきである．

3.2 ネットワーク層の安全性に対する脅威

情報の非一貫性と不正確さの問題は，様々な層での安全性に影響を及ぼす．しかし，一般に，ネットワーク層より上位の層ではそれぞれのアプリケーションにおいて通信相手どうし (end-to-end) での対策がとられるべきであろう．そのような対策により，多くの下位層レベルでの攻撃の意味がなくなる．たとえば，上位層での暗号化により下位層での情報監視の意味はほとんどなくなってしまう．しかし，そのような上位層での対策がとられても，なお，ネットワーク層における安全性に対する脅威は存在する．たとえば，前節と同様なインターネット環境では，次のようなことが起こりうる：

(1) 改ざん
(2) ネットワーク資源不正使用
(3) トラフィック生成による上位層サービス拒否
    a. 連鎖反応による網の嵐攻撃
    b. 目標ホスト宛の多量トラフィック生成攻撃
(4) 無認可のネットワーク情報公開

改ざん攻撃では，経路情報交換プロトコルにより不正情報を流し，パケットが偽ルータへ転送されるようにする．パケットは偽ルータで内容の改ざんが行われ，正規ルータへ転送される．偽ルータは使用されていないネットワークアドレスを使い，経路情報交換プロトコルに参加する．あるいは，管理プロトコルを使い，ホストの経路表を修正させることも可能である．それを受け取ったホスト群は，以後，その偽ルータへパケットを送る．こうして，偽ルータは，これらのホストからのトラフィックを自由に制御できるのである．このように改ざんの機会を得た偽ルータは，パケットの発信元アドレスを自分のアドレスに設定し，正規の発信元と宛先の間の通信に介在できる．アプリケーション層での暗号化などの安全対策がとられていても，偽ルータでは受け取ったパケットの一部を故意に落とすような形での情報改ざんも可能である．

ネットワーク資源不正使用攻撃では，未使用のネットワークアドレスを偽ホストに設定し，正規の登録手続きをふまずにネットワークを使用し，他の偽ホスト群と通信し合う．ネットワーク資源の不正使用が重大な脅威となるかどうかは，組織間基幹ネットワークや組織内の LAN など，ネットワーク環境の特性と，不正使用によるコストの全体に及ぼす相対的な大きさにより決定されるだろう．したがって，重大さは各サブネットのメディアの性質などによって異なるので，その対策はサブネットごとに解決されるべきであろう．

トラフィック生成によるサービス拒否攻撃には，2種類ある．前節で述べた連鎖反応を使用したものと，多量のパケット群を被害者ホスト宛に向け発信するものである．前者では，偽ホストを下位層プロトコルの放送アドレスに設定し，そのホストから自分自身宛のパケットを発信することで，そのサブネット上のルータの数によって，網の嵐を起こすことが可能である．また，発信されるパケットの宛先がそのサブネットの外のものだとすると，サブネット内に網の嵐を起こすだけでなく，外部へ流す余分なトラフィックを流してしまう．

後者のトラフィック生成攻撃では，偽ホストから一方的にあるホストあるいはホスト群に向けてトラフィックをネットワーク上に発生させる．ノードがルータやサーバの場合，そのネットワークインタフェースに輻輳を起こすため，ネットワークのユーザが受ける影響は大きい．偽ホストは自分のネットワークアドレスや下位層アドレスを時によって変えてしまうであろうから，偽ホストの特定は不可能に近い．

ネットワーク情報の不正監視は，ネットワークのト

ラフィックを監視することで，他に気づかれずに他のノードのアドレスや非一貫性などのネットワーク構成情報が読み取られ，攻撃者に役立つ情報入手の機会を与えてしまう．

4. アドレス認証による対策

4.1 概　　要

構成発見の延長上には前章で述べたような問題が存在し，それらの解決なしには，真の構成発見の問題解決はありえない．しかし，これらすべての問題に対する対策を行おうとすると，それぞれの対策の最大公倍数になってしまい，膨大な制御システムの構築が必要となる．したがって，その環境に応じた対策をとるべきである．

対策を立てるより，監視装置で十分ではないかという見方もできる．しかし，地球規模に広がったインターネット環境などをみると，すでに各ホストやルータさえも，熟練した管理者の制御下にあることは少なくなり，また，そのような管理者がいる場合にはその求められる管理範囲が大きく，監視装置からの情報の有効な活用を人間レベルの管理者に要求することは難しくなりつつある．また，パーソナル・コンピュータの普及にともない，「接続即稼働」（plug and play）と呼ばれるような，より安易なネットワーク接続の方向へと進んでいる．このような状況では，ネットワーク自体が解決していくようなメカニズムが望ましい．

これらの観点から，本論文では，ネットワーク管理の面から波及効果の度合が大きいとみられる設定ミス，あるいは故意による誤った下位層アドレスの問題と経路制御改ざんの2つの問題について，アドレス情報についての認証による解決を提案する．ここで紹介するサブネット内の対策では，アドレス変換プロトコルで通常のメッセージに証明書を付け，サブネット内では証明された情報だけを使用するようにする．証明書とは，ネットワークアドレスから下位層アドレス（MACアドレス）の変換を示すもので，その設定が検証された時点で公証機関（Certification Authority: CA）から発行されたものである．この対策は，結果的にみると，資源の不正使用にも有効である．ただし，この対策では，目標ホスト宛の多量トラフィック生成攻撃や，無認可のネットワーク情報公開などの脅威は解決できない．

この対策の実施には，公証機関からサブネット上の各ホストやルータなどのノードへの証明書の発行手順が必要となる．このために，現状のアドレス情報の流れに次の2つの処理手順を追加する．1つは，オフラインでのノード情報の登録で，もう1つはノードがネットワーク上での活動を始めた時点でのその設定の検証である．検証結果が，事前に登録されたノードの情報と一致したときに，公証機関は，証明書をノードに対して発行し，情報システムは証明された情報だけを保持する．ネットワーク上のノードは，情報システムから他のノードについての情報を得て動作するので，検証された情報のみがネットワーク上で使われるようになる．

以下，4.2節で証明書をサブネットのアドレス変換プロトコルで使用する対策がどのように下位層アドレスや不正な経路制御による改ざんなどの問題を解決するかを論じる．4.3節では改良されたアドレス情報の流れを説明し，4.4節ではネットワーク上での証明書発行手順を紹介する．4.5節では本解決法の特徴を述べ，4.6節ではアドレス変換プロトコルでの証明書の使用による負荷を吟味し，4.7節ではこのような証明書の利用可能な他のアプリケーションを紹介する．

4.2 アドレス変換プロトコルでの証明書使用による対策

コンピュータ通信ネットワーク上のホストが他と通信したいとき，相手のホストの名前をネーム・サーバに照会して，宛先のネットワークアドレスを得る．発信ホストは，そのネットワークアドレス先にはどうやって到達すればよいかを経路表に照会する．相手が同じネットワーク上にあれば，アドレス変換表を使って，相手のネットワークアドレスから下位層アドレスを引き出す．相手が異なるサブネットにある場合，次に転送すべきサブネット上のルータのネットワークアドレスを経路表から得，アドレス変換表により，そのルータの下位層アドレスを得，パケットをルータへ転送する．このアドレス変換表の情報を維持するのがアドレス変換プロトコルである．

現在インターネットで使われているアドレス変換プロトコル（ARP）[9]は，もともと，MITのArtificial Laboratoryで開発されたLAN，ケイオスネット（Chaosnet）[13]のために作られたプロトコルだが，イーサネット上で広く使われている．非コネクション型で，問合せ（Request）と回答（Reply）のメッセージの組合せにより，各ノードが他のノードの下位層アドレス情報を得ることができる．問合せには，知りたい相手のネットワークアドレスを提示し，送り手のネットワークアドレスと下位層アドレスも付けて，サブネット全体に放送する．他のノードは，この問合せが自分のことであれば，回答メッセージを先の送り手に返し，下位層アドレスを教える．

このような方法だけでは，n 個のノードがサブネット上に存在した場合，最多 $n(n-1)$ 個の問合せがされるはずである．しかし，実際には，各ノードは，放送される問合せメッセージから，送り手のネットワークアドレスと下位層アドレスの組合せを学習し，内部のアドレス変換表に入れておく．この変換表の各エントリーは普通 20 秒くらいで消去される．したがって，つねにある程度の ARP メッセージはサブネット上に現れる．

本論文でいう証明書は，アドレス変換プロトコル用に，ノードのネットワークアドレスとサブネットの下位層アドレスの組合せを公証機関（CA）が証明したものである．証明には公開鍵システム[14]を用い，証明書はアドレス情報を日時のスタンプ加え，CA の秘密鍵で暗号化したもので，以下のような形式である．

$\{\text{CA'sID}, \{\text{下位層アドレス}, \text{ネットワークアドレス}, \text{日時}\}SK_{CA}\}$

公開鍵システムでは秘密鍵と公開鍵という 2 種類の鍵が存在し，秘密鍵で暗号化したものは，公開鍵で復号することができ，また，逆に公開鍵で暗号化したものは秘密鍵で復号することができる．秘密鍵は 1 人しか知らないが，公開鍵は他の多くの人たちが知っている．したがって，CA の秘密鍵で暗号化されたものは，他のノードは広く知られている公開鍵で復号することができ，またその暗号化された情報は確かに CA が発信源だということを信じることができる．

証明書は各ノードの設定が正しいかどうか検証されてから発行されるものとする．アドレス変換プロトコルの問合せや回答の各メッセージに発信者の証明書を付けることにより，サブネット内では，証明された情報だけがアドレス変換表に現れるようになる．無為か故意による下位層アドレス問題は，ノードの下位層アドレスの設定を検証する段階で発見され，現状のアドレス設定についての証明書は発行されない．したがって，他のノードのアドレス変換表には現れないので，正規のノードからそのアドレスへパケットが送られることはない．

承認されたノードだけが，証明書を与えられ，登録されていないネットワークアドレスを使った偽ルータは証明書がないので，不正な経路制御による改ざんの問題についても証明された情報を使用することで防ぐことができる．

4.3 現在のアドレス情報の流れの修正

現在，ネットワークのアドレス情報は次のような 4 段階を経て，ネットワーク上に現れる．

(1) 資源承認（Resource Admission）

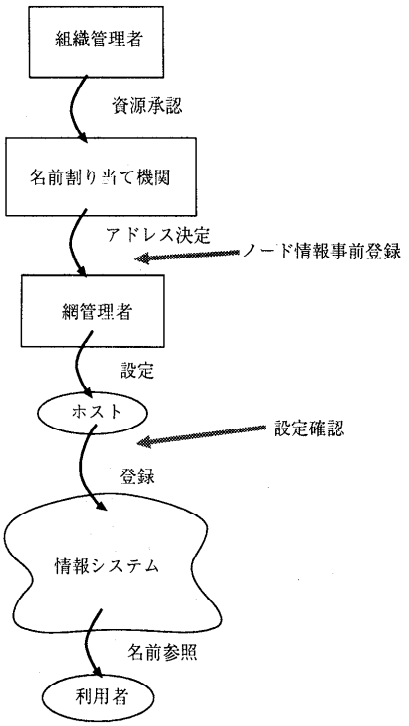


図 2 アドレス情報の流れ
Fig. 2 The flow of address information.

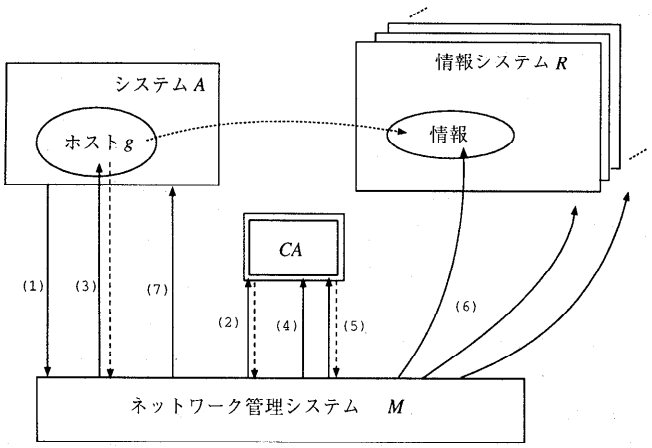
(2) アドレス決定（Naming）

(3) 設定（Configuration）

(4) 登録（Registration）

図 2 はこれらの流れを示す．**資源承認**では，組織の方針によって，ホストやルータなどをネットワークに加えるかどうかを決定する．承認されたホストやルータは**アドレス決定**で，ネットワークアドレスが割当てられ，**設定**により，ネットワークへ接続される．接続されたホストやルータは様々なネットワーク動作により自分のアドレスを他へ知らせていく．それを**登録**と呼ぶ．

さて，ここで現在のこのアドレスの流れに新たに 2 つの手順を加え，証明書発行を可能にする．それは，オフラインでの**ノード情報事前登録**と**設定確認**である．アドレス決定の後，アドレスとそのノードの暗号システムの属性，すなわち，公開鍵などの情報を登録するのが，**ノード情報事前登録**である．この事前登録が完了すると，ホストやルータに対して登録番号（registration ID）が発行される．**設定確認**では，ホストやルータがネットワーク活動を開始した時点で，登録番号を使いながら，ネットワーク管理システムから事前登録したとおりのアドレスなどの正しい設定が



(1): システム A が登録番号を提示してアドレス登録を開始する
(2): M が CA に登録番号についての情報を問い合わせる
(3): 情報に基づいて M は g の設定を検証する
(4): M は検証結果を CA に報告する
(5): M は CA に証明書の発行を促す
(6): M は情報システムにアドレス証明書を置く
(7): M は A にアドレス証明書を届ける

図3 登録プロトコル
Fig. 3 Our registration protocol.

されているかどうかを検証され，確認がとれた段階で証明書を受け取る．その後，証明書付きの情報が登録されるのである．次節でこの設定確認と登録をネットワーク上で行う登録プロトコルを説明する．

4.4 登録プロトコル

登録プロトコルは次のようになる．物理的なワークステーションのようなシステムがネットワーク上のホストやルータとして動作するためには，ネットワークインタフェースとそれを動作させるソフトウェアの導入，またそれらの諸パラメータの設定が必要である．これらの設定作業が終了すると，そのホストあるいはルータはネットワークへ接続される．登録プロトコルでは，ホストやルータは登録番号をネットワーク管理システムに提示して，設定確認およびアドレス登録を始める．登録番号は，ネットワーク上では当該ホストあるいはルータと公証機関だけが知っている共有秘密であり，設定確認のセッション ID の役割を果たす．

ネットワーク管理システムはその登録番号についての情報を公証機関に問い合わせる．公証機関からの情報に基づいて管理システムは設定確認を行う．設定確認では，アドレスを含む諸パラメータ設定と，割り当てられたアドレスがネットワーク上で実際に使用可能であるかの確認を行う．設定確認には，一般的なネットワーク管理用の問合せと返答の組合せのメッセージを使うので，同じアドレスのホストがすでに存在していれば，設定確認の問合せに対し，既存のホストも答えようとするので確認できる．確認結果は，公証機関に報告され，それに基づいて，公証機関は当該ホストあるいはルータにアドレス証明書を発行する．証明書は管理システムに渡され，そこから，ネーム・サーバやディレクトリなどのデータベース型の静的情報システムに直接登録されるか，またはホストやルータ自身から証明された情報が学習型の動的情報システムであるアドレス変換システムに登録される．

図3 はこの登録プロトコルの概要を示す．

この登録プロトコルで重要な点は，登録番号を使ってホストやルータの認証だけで終わらずに，設定を確認したうえで，証明書を発行していることである．**認証**とは一般に通信の参加者が真にその参加者自身であるかどうかを確認することである．しかし，本論文では，割り当てられたアドレスが確かにホストやルータに設定されているか，という設定情報の認証手続きが目的であるので，設定確認を必要とする．

4.5 本解決法の特徴

MIT の Kerberos[15]は，単なるユーザやホスト自身の認証[16]だが，本論文で示した解決法は，ホストの情報の認証である．ホストが起動した時点で行われるオンライン登録の際の設定確認はそのために必要であっ

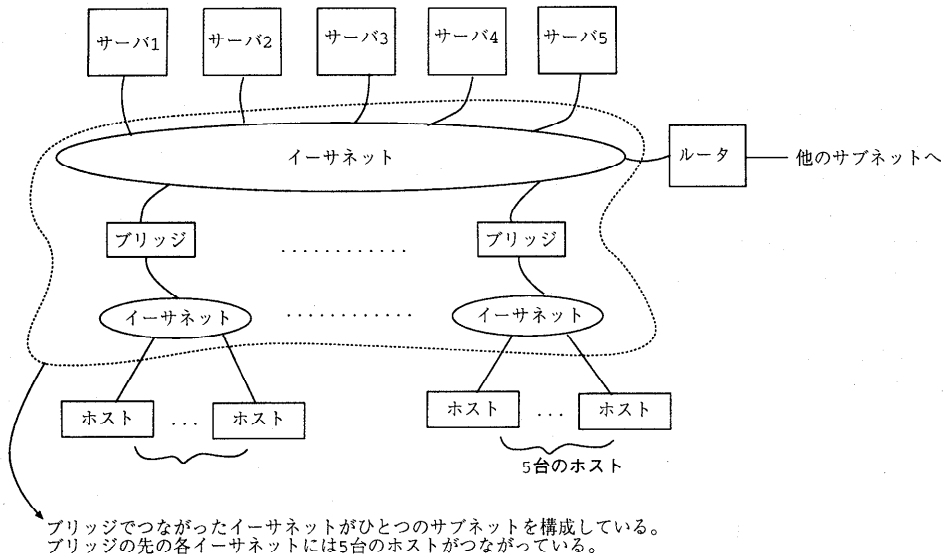


図4 仮定したサブネット環境
Fig. 4 An example environment.

た．すなわち，ホストについての情報と実際の構成に違いがないかが検証された．証明書（certificate）は，後で情報のユーザがその情報の正しさを認証するうえで必要である．

本研究の目的は情報の完全性（integrity）を保持することであるので，この登録手順では秘匿性は保たれない．しかし，秘匿性が必要とされる状況では，各メッセージを受信者の公開鍵で暗号化することができる．あるいは，管理手順のメッセージについて同じ共通鍵を使うこともできるであろう．

X.500ディレクトリの公証機関（CA）は単に証明書を発行する機関だが，本解決法のCAは，ネットワーク管理システムとの連携により，情報が事実と検証された時点で証明書を発行する．したがって，本CAは，検証された情報に対してしか証明書を発行しないという点から，X.500のCAよりもさらに大きな権威を持つ．そのうえ，本CAはオフラインでの登録にも関わっているので，X.500のCAにくらべ，能動的に証明作業に携わっている．

4.6 アドレス変換プロトコルでの証明書使用による負荷

本節では，4.2節で示したような，イーサネットでのアドレス変換プロトコル環境における証明書使用を想定した場合の負荷について吟味する．

証明書使用の場合，既存プロトコルの各メッセージに発信者のアドレスについての証明書を付加する．証明書の形式は4.2節で述べたとおりである．既存のARPメッセージは28バイト長であるが，証明書の64バイト分が追加される．実験ではRSA[17]をソフトウェアで実装したものを使用し，512ビット長の公開鍵を使用した．またハードウェアはSUN3/50とSUN4/64を使用した．前者では，平均で証明書発行のためのCAの秘密鍵による暗号化には，平均で4.55秒かかり，受信者による証明書のCAの公開鍵を使っての展開には0.41秒かかった．また，後者では，暗号化に2.17秒，展開に0.20秒であった．すなわち，暗号化には数秒のオーダの時間がかかるが，復号についてはその10分の1であった．アドレス変換プロトコルでは復号だけ必要である．すなわち，アドレス変換プロトコル動作で，新たに学ぶ下位層およびネットワークアドレスについて既存のものに比べ，数百ミリ秒余計な負荷がかかる．

証明書の復号作業の頻度の調査を次のような仮定で行った．1つのサブネット上には100台のノードがあり，その中に5台のサーバが存在し，各ホストは4台ずつの分散動作の仲間（peer）があり頻繁にやりとりがあり，5台のサーバのうち2台を利用すると仮定した．このネットワーク形態は，実験当時，ロンドン大学内University College Londonの理学部計算機科学科内部の典型的なサブネット環境に基づいて決めたものである．図4に仮定した環境を示す．すべてのノードがそれぞれ5台の仲間と2台のサーバについてのアドレス変換の情報を得るためには，341組の問合せと回答が必要である．いったんつくられた変換エント

リーは，放送される問合せメッセージを監視することにより，更新されていくので，165組のやりとりで十分である．実際には，前出の学科内サブネットを観測してみると，アドレス変換表の各エントリーの最長記憶時間（time out）20分の間に，150組前後のやりとりがあった．すべてのホストに監視される問合せメッセージの抽出平均到着間隔は8秒であった．

SUN4でアドレス証明書の復号を行い変換表のエントリーに入れる作業に平均0.3秒かかった．したがって，アドレス変換プロトコルのメッセージの到着間隔が数秒のオーダであれば，証明書つきアドレス変換表操作の時間はずっと短い．

実時間のアプリケーションは，エントリー生成の場合にのみ，その変換表操作のため，数百ミリ秒を待たなければならない．もし，この待ち時間がアプリケーションの支障をきたすという場合には，2つの変換表を持たせることができよう．一方は既存のメッセージ部分から学んだ情報で形成し，他方は証明書の情報で形成する．ホストは通信の際，後者の変換表を先にみて，そこにないものについては前者の変換表を使用する．すなわち，各ホストは数百ミリ秒だけの間安全ではないかもしれない変換表の下に動作するのである．

このように，証明書を付加することにより，変換表操作の時間の数百秒の遅延は起こるが，網の嵐や改ざんなどの脅威は回避できる．構成が頻繁に変わることのない環境では，変換表のエントリーの時効時間を数時間などに長くできよう．こうすると，たとえば，前出の頻度調査の仮定例の環境では，数時間ごとに2分（341 × 0.3 秒）の初期遅延だけが必要となる．

4.7 証明書の応用

証明書の応用は，使用されるアプリケーションごとに必要な内容が異なるが，本研究で提案する発行手続きを使用できる．応用できるアプリケーションとしては，データベース型の情報システムでは，ドメイン名変換システム（DNS）[6]や，X.500ディレクトリ[7]がある．また，アドレス変換プロトコルと同様に，学習型の情報システムである経路制御情報交換システムや様々な学習および発見システムなどへの応用があげられる．

DNSの場合，たとえば，証明書の内容はホスト名とIPアドレスの組合せとする．学習型では，経路上で転送可能パケットの大きさの最小値，すなわち最小MTU（Maximum Transmission Unit）を見つけるためのMTU発見（MTU Discovery）[22]などでも応用できる．この場合，証明書内容には，ネットワーク・アドレス，ネットワーク・アドレス中のネットワーク番号を抽出するためのネットワーク・マスク，およびMTUサイズなどが含まれなければならない．また，ネットワーク管理システムでは，アドレス変換に利用した証明書をそのままサブネット内のホスト認証のために利用することも可能である．経路情報交換システムでは，通常，同じサブネット内のルータどうしで隣接ノード確認が行われるので，今回アドレス変換に使用した証明書がそのまま認証のために利用できる．

5. むすび

インターネットは様々な種類のネットワークの相互接続性を重視して設計されてきた背景があるため[1),2)]，それに相反するアクセス制御は採り入れられにくい．したがって，現在のインターネット環境は安全とはいいがたい．様々な落し穴的状態[18)]や，オペレータのタイプミスなどの小さな間違いからネットワーク運用に大きな影響を与える場合があっても不思議ではない．過去には，インターネットが，研究機関だけをつなげていた事情や，ネットワーク全体の規模が小さく情報が熟練したオペレータによって管理されていた事情から，起こりうる脅威は避けられてきた．インターネットの拡大と各ノードの多様化にともない，情報管理者の熟練度の低下などが起こり，今，ネットワーク自体が無為あるいは故意による脅威を回避していく形が求められている．

本論文では，構成発見という新しい視点から，構成情報の重要な要素であるアドレス情報に着目し，情報の不正確さから起こる脅威を論じ，その対策として情報認証を提案した．情報の登録に至る流れを安全性の観点から完全なものにするために，情報と現状の検証を行い，さらに公開鍵方式の暗号化の手法を使い，検証された情報についての証明書を発行する手順を示した．この登録手順については，すでにBANロジック[19)]を使用して情報の流れの完全性を証明した[20)]．

さらに，アドレス変換プロトコルでの情報証明書の応用を考察した．興味深い点として，証明書認証のための公開鍵による暗号の解読は，アルゴリズムにもよるが，今回の実装においては，証明書作成のための公証機関の秘密鍵による暗号化の10分の1で済むという事実があげられよう．これは，発行時間よりも使用時における解読時間の短さを必要とする証明書のネットワーク運用での応用に則しているといえよう．

他に証明書を利用できるアプリケーションには，ドメイン名変換システム[6)]や，X.500ディレクトリ[7)]，経路制御情報交換システムや様々な学習および発見システムなどへの応用が考えられよう．

本論文では，特にアドレスの入力手順に的をしぼったが，証明書の利用については，ネットワークから外されたノードのアドレスの証明書を無効にすることなど，有効性の管理は今後の課題である．

本論文では，証明書の発行手順に重点を置いているので，この既存の証明書の保持については将来の研究事項としておく．

今回は，情報の不正確さに焦点をあてて，議論を進めたが，構成発見問題では，一貫性の問題はどの情報システムでも問題である．確かに，今回の対策でも，つねに証明書を発行されたシステムを監視するなどの対策も可能であるが，エレガントな解ではない．今後，一貫性問題に着目し，何らかの認証手法を応用することを試みたい．

本研究は，ネットワークでの情報証明の必要性であったが，今後，構成発見の他分野における問題と情報証明の応用を試みたい．特に，最近，データベース分野において，知識発見（Knowledge Discovery in Databases）[21]が研究され始めたが，この分野における情報証明は興味深い．

また，複数の公証機関の信頼関係による証明書の利用の考察も必要である．たとえば，ある組織で発行された証明書は，他の組織内ではその組織の公証機関の発行公証機関に対する信頼度により利用可能か否かが決まる．X.509 の信頼連鎖（chain of trust）などを参考に，この分野の研究が必要である．

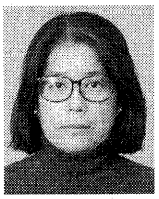
謝辞 ロンドン大学内 University College London における本研究に際し，Peter T. Kirstein 教授の指導および支援をいただき，感謝します．実験に際しご協力いただいた Peter Williams, John Andrews, Ping Hu の諸氏に感謝します．Radia Perlman 氏との電子メイルでの会話は，本論文の誤った下位層アドレスの問題の考察に非常に役立ちました．また，匿名査読者の方々の貴重なコメントに感謝いたします．

参考文献

1) Pouzin, L.: Internetworking, *Computer Communications*, Vol.2, Chapter 15, Prentice-Hall (1985).
2) Cerf, V.G. and Cain, E.: The DoD Internet Architecture Model, *Computer Networks*, pp.307–318 (1983).
3) Cerf, V.G. and Kirstein, P.T.: Issues in Packet-Network Interconnection, *Proc. IEEE*, Vol.66, No.11, pp.1386–1408 (1978).
4) Berners-Lee, T., Cailliau, R., Luotonen, A., Nielsen, H.F. and Secret, A.: The World-Wide Web, *CACM*, Vol.37, No.8, pp.76–82 (1994).
5) Shoch, J.: Inter-Network Naming, Addressing, and Routing, *Proc. IEEE COMPCON Fall 1978*, pp.72–79 (1978).
6) Mockapetris, P.: Domain names-concepts and facilities, *Request for Comments*, No.1034 (1987).
7) CCITT and ISO: Recommendations X.500 Series; the Directory, International Standard (1988).
8) Rosen, E.C.: Exterior Gateway Protocol (EGP), *Request for Comments*, No.827 (1982).
9) Plummer, D.: An Ethernet Address Resolution Protocol, *Request for Comments*, No.826 (1982).
10) Perlman, R.: *Interconnections: Bridges and Routers*, Addison-Wesley (1992).
11) Cheriton, D.R. and Mann, T.P.: Decentralizing a Global Naming Service for Improved Performance and Fault Tolerance, *ACM Trans. Computer Systems*, Vol.7, No.2, pp.147–183 (1989).
12) Manber, U.: Chain Reactions in Networks, *IEEE Computer*, Vol.23, No.10, pp.57–63 (1990).
13) Moon, D.: Chaosnet, *A.I. Memo*, No.628, Artificial Intelligence Laboratory, MIT (1981).
14) Hellman, M.: Commercial Encryption, *IEEE Network Magazine*, Vol.1, No.2, pp.6–10 (1987).
15) Steiner, J.G., Neuman, B.C. and Schiller, J.I.: Kerberos: an Authentication Service for Open Network Systems, *Proc. USENIX Winter Conference*, pp.183–190 (1988).
16) Needham, R. and Schroeder, M.: Using Encryption for Authentication in Large Networks of Computers, *CACM*, Vol.21, No.12, pp.993–999 (1978).
17) Rivest, R., Shamir, A. and Adleman, L.: A Method for Obtaining Digital Signatures and Public-Key Cryptosystems, *CACM*, Vol.21, No.2, pp.120–126 (1978).
18) Bellovin, S.M.: Security Problems in the TCP/IP Protocol Suite, *ACM Computer Communication Review*, Vol.19, No.2 (1989).
19) Burrows, M., Abadi, M. and Needham, R.: A Logic of Authentication, Technical Report, DEC Systems Research Center, No.39 (1989).
20) Murayama, Y.: Using BAN Logic for the Proof of Network Address Registration Protocol, *Trans. IPSJ*, Vol.37, No.5, pp.779–789 (1996).
21) Fayya, U. and Uthurusamy, R.: Data Mining and Knowledge Discovery in Databases,

CACM, Vol.39, No.11, pp.24–26 (1996).
22) Mogul, J. and Deering, S.: Path MTU Discovery, *Request for Comments*, No.1191 (1990).

(平成9年 5 月12日受付)
(平成9年12月 1 日採録)



村山 優子（正会員）

津田塾大学学芸学部数学科卒業．三菱銀行および横河ヒューレット・パッカード社に勤務．昭和59年University College London大学院理学部計算機科学科修士課程修了．平成2年同大学大学院博士課程修了．Ph.D.（ロンドン大学）．慶應義塾大学環境情報学部非常勤講師を経て，平成6年4月広島市立大学情報科学部情報工学科講師．現在に至る．インターネット，ネットワークセキュリティの研究に従事．著書『ネットワーク概論』（サイエンス社）．IEEE，ACM，電子情報通信学会，映像情報メディア学会，日本OR学会，情報知識学会会員．