# IDENT代理サーバによるリモートアクセスユーザ認証機構

山井成良[1]，中西透[2]，安倍広多[3]，石橋勇人[3]，松浦敏雄[3]，岡本卓爾[2]，

[1]岡山大学 総合情報処理センター
[2]岡山大学 工学部
[3]大阪市立大学 学術情報総合センター

**概　要**

IDENTプロトコルはアプリケーションレベルでのユーザ認証の手段として広く用いられている．しかし，リモートアクセス環境ではユーザの計算機上でIDENTサーバが動作していない，あるいは動作していてもIDENTサーバの返す情報が信頼できないなどの理由により，IDENTプロトコルを用いることができなかった．本論文では，IDENT代理サーバを導入し，ユーザの計算機に代わってネットワーク接続時の認証情報を返すことによりアプリケーションレベルでのユーザ認証を行う方法を提案する．また，本方法の実装や運用試験を行った結果についても報告する．

# An Authentication Mechanism for Remote Access Users with an IDENT proxy server

Nariyoshi Yamai[1], Toru Nakanishi[2], Kota Abe[3],
Hayato Ishibashi[3], Toshio Matsuura[3] and Takuji Okamoto[2]

[1]Computer Center, Okayama University
[2]Faculty of Engineering, Okayama University
[3]Media Center, Osaka City University

**Abstract**

IDENT protocol is one of the most popular user identification methods on application layer. However, this protocol is not used on remote access environment because any IDENT server on user's computer is neither available nor trusted. In this paper, we propose an user identification method available even on remote access environment, by introducing an IDENT proxy which responds the user name authenticated on dial-up connection. Then, we also present our implementation and empirical results.

## 1　まえがき

近年，ネットワーク接続事業者の急速な増加や高速モデムなどの発達に伴って，ユーザが遠隔地から公衆電話網を経由して所属組織内のネットワークにアクセスするような利用形態（リモートアクセス）が急速に普及してきている．このような利用形態では，ユーザの計算機が組織内のネットワーク管理者ではなくユーザ自身の管理下に置かれており，また，ネットワーク管理者から離れた場所に設置されているために，様々な不正利用が発生している．中でも特に，電子メールでのアドレス詐称や本来利用できないサービスでの偽のユーザ名を騙った利用などといったアプリケーションレベルでの不正利用が頻発している．このため，リモートアクセス環境におけるアプリケーションレベルでの不正利用を防止・抑制するための簡便な方法の開発が強く要求されている．

ネットワークにおける不正利用を防止・抑制するための有力な方法として，ユーザ認証が広く用いられている．リモートアクセス環境におけるユーザ認証には，PAP[1]，CHAP[2]などがあるが，これら

はダイアルアップ接続時の認証を目的としたプロトコルであり，アプリケーションレベルでのユーザ認証には利用できない．他方，アプリケーションレベルでのユーザ認証の代表的な方法として，ディジタル署名 [3, 4] や IDENT プロトコル [5] を利用した方法が知られている．ディジタル署名による方法は厳密な認証が期待でき，原理的にリモートアクセス環境にも適用できるが，ユーザの計算機側での秘密鍵の管理のために負担が大きくなりすぎて，様々な種類の計算機が接続されるリモートアクセス環境に対して手軽に適用することはできない．それに対して，IDENT プロトコルによる方法は，TCP に基づいたアプリケーションのユーザ認証に限定されているものの，ユーザが秘密鍵などの秘密情報を管理する必要がないので，ユーザの計算機での負担は小さく，これまでに広く利用されている．しかし，ユーザの計算機がリモートアクセス環境にある場合には，これをリモートホストとみなして IDENT サーバを実装したとしても，ネットワーク管理者の管理外にあるので，これが返すユーザ名は信頼できない．すなわち，この方法はそのままではリモートアクセス環境に適用できない．

本論文の目的は，リモートアクセス環境の下で，ユーザの計算機に大きな負担を強いることなく，アプリケーションレベルでのユーザ認証を簡便に行う方法の開発にある．この目的を達成するため，我々はネットワーク管理者の管理下にある計算機上に IDENT 代理サーバを導入し，IDENT プロトコルによる問い合わせに対して，IDENT 代理サーバがダイアルアップ接続時の認証結果を利用してユーザ名を応答する方法を提案する．この場合，問い合わせに対して IDENT 代理サーバが無条件にユーザ名を応答すると，任意の計算機から不正にユーザ名を取得できる点が問題となる．この問題に対処するため，IDENT 代理サーバは問い合わせを受けた TCP コネクションが実際に存在するかを確認し，存在した場合だけユーザ名を返すようにしている．

## 2 リモートアクセス環境

リモートアクセス環境においては，多くの場合，ユーザが接続先ネットワークへのアクセス権を有するか否かを調べるために，既に述べたダイアルアップ接続時の認証が行われる．そこで本論文では，ユーザ認証の対象とするリモートアクセス環境として，図 1のようなネットワーク構成を想定する．

破線で囲んだ部分は組織内のネットワークであり，その組織のネットワーク管理者の管理下にある．RAS 及び AS は，それぞれ，リモートアクセスサーバ及びダイアルアップ接続用の認証サーバである．また，LN はこれ以外の組織内のネットワークで，このネットワークには種々のサービスを提供する計算機群が存在する．他方，破線外にある $RC_1, RC_2, \ldots, RC_n$は，遠隔地のユーザが利用する $n$ 台のリモートアクセス計算機であり，それぞれユーザ自身の管理下にある．$RC_i (1 \leq i \leq n)$ には前述したダイアルアップ接続時の認証のためのソフトウェアが実装されているが，その種類や搭載されている OS は任意である．なお，LN から外部へ向かう実線は，広域ネットワークへの接続を示す．

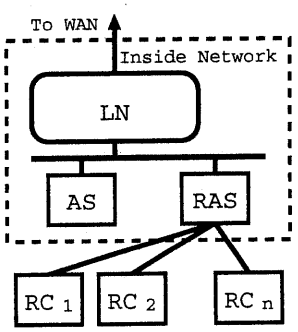


図 1: リモートアクセス環境におけるネットワーク構成

ここでは，以降の議論を明確にするために，さらに以下の仮定を設ける．

(a) $RC_i$を利用するのは，ダイアルアップにより接続してからこれを切断するまでの間，同一のユーザである．

この仮定では，ユーザ A がダイアルアップ接続時の認証を受けた後，ある別のユーザ B が LN に不正アクセスするという可能性が残されている．しかし，この場合，B による不正利用の責任は，最終的にはこのような利用を許した A に帰すると考えられる．従って，このような仮定を設けたとしても，実用上差し支えないと考えられる．

次に，この環境におけるユーザ認証について以下のように想定する．

LN に接続されている計算機では，TCP によりサービスを提供するサーバプログラム SP が動作し

ており，IDENTプロトコルによる認証を行う．SPとそのクライアントプログラムCPとの間にTCPコネクションCが確立されており，またCPが動作している計算機上でIDENTサーバISが動作しているものとする．このとき，SPはIDENTプロトコルを用いて以下の手順によりCPのユーザを認証する．

(1) SPは，Cの相手側のIPアドレスとポート番号を取得し，このアドレスを用いてISとの間にコネクションを確立する．

(2) SPはISにCの両端のポート番号を送る．

(3) ISはSPの動作している計算機のIPアドレスを求め，このアドレスと自計算機のIPアドレス，並びにCの両端のポート番号からCが存在するかどうか確認する．

(4) もしCの存在が確認されれば，Cの所有者名(CPのユーザ名)を応答する．そうでなければ，エラーメッセージを応答する．

ここで，IDENTプロトコルではCの両端のIPアドレスはSP–IS間のコネクションから求められるため，問い合わせの際には両端のポート番号だけがISに送られることに注意する．

以下本論文では，このようなリモートアクセス環境を対象に，$RC_i$のユーザ認証を考える．

# 3 リモートアクセス環境におけるユーザ認証

## 3.1 ユーザ認証システムの構成

$RC_i$のユーザ認証を行うために，本論文では1.で述べたように，ネットワーク管理者の管理下にある計算機上にIDENT代理サーバを新たに設ける方法を提案する．この方法では，仮定(a)から，接続時の認証結果をネットワーク管理者の管理下で保存し，これを利用して，IDENT代理サーバがIDENTプロトコルの問い合わせに応答すればよい．

本論文では，このようなユーザ認証システムの構成にあたって，仮定(a)に加えてさらに以下の仮定を設ける．

(b) $RC_i$がSPと通信する際，IPアドレスやポート番号の偽造などトランスポート層やネットワーク層での不正はない．

(c) ダイアルアップにより接続してからこれを切断するまでの間，$RC_i$のユーザ名と$RC_i$に割り当てられたIPアドレスの組がASに保存されている．

このうち，仮定(b)については，リモートアクセス環境ではRASを介して通信を行うため，トランスポート層やネットワーク層で不正を行うとRASが中継できず，この仮定を満たすと考えられる．また，仮定(c)については，例えばRADIUS[7, 8]など，多くの認証サーバは接続記録を残すことが可能であるため，この仮定を満たすことは困難ではない．以下，ASに保存されている$RC_i$のユーザ名を$U_i$，$RC_i$に割り当てられたIPアドレスを$AR_i$と表記する．

IDENTプロトコルを利用しているサーバプログラムはNS内の計算機に数多く実装されているため，$RC_i$からの利用に対してプログラムを変更しないでユーザ認証できることが望ましい．しかし，図1の構成のままでは，IDENTプロトコルによる問い合わせは$RC_i$に対して行われてしまう．そこで図2に示すように，RASとLNの間に新たにルータRを導入し，$RC_i$のIDENTサーバへの問い合わせをIDENT代理サーバIPSに転送させるようにする．このとき，IPSはSPから$RC_i$のユーザ名の問い合わせを受けると，ASからこのユーザ名を取得し，これをSPに応答する．

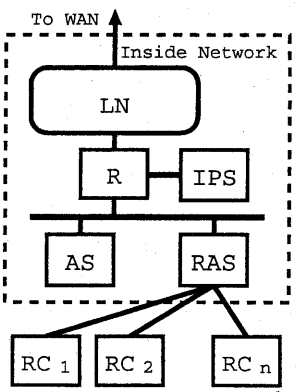


図2: リモートアクセス環境におけるユーザ認証を行うためのネットワーク構成

### 3.2 ルータとIDENT代理サーバの動作

図2の構成では，SPから$RC_i$のIDENTサーバへの問い合わせをIPSに転送する機能がルータRに必要である．このような機能として，本論文ではアドレス変換機構(Network Address Translator[6]: 以下，NATと呼ぶ)を用いる．

このとき，単純に送信元あるいは送信先のIPアドレスだけを変更すると，IPSにはコネクションの両端のポート番号しか送られないため，どのリモートアクセス計算機に対する問い合わせかをIPSが特定できない点が問題となる．そこで，これを解決する一方法として，各$RC_i$に対してIPS側で個別にポートを用意し，RでIPアドレスだけでなくポート番号も変更するようにする．すなわち，IPSのIPアドレスをAI，$RC_i$に対するIPS上のポート番号を$PI_i$とすると，Rでは以下のような処理が行われる．

(1) 送信先IPアドレス，送信先ポート番号がそれぞれ$AR_i$，113(IDENTのポート番号)であるパケットを受信すると，それぞれをAI，$PI_i$に変換して中継する．

(2) 送信元IPアドレス，送信元ポート番号がそれぞれAI，$PI_i$であるパケットを受信すると，それぞれを$AR_i$，113に変換して中継する．

別の問題として，IPSが問い合わせられたコネクションの存在をどのように確認するかがある．すなわち，図2の構成ではSPが動作する計算機と$RC_i$との間の通信はIPSを経由しないため，IPSは単独では両者間でどのようなコネクションが確立されているかを知ることができない点が問題となる．これに対処する方法として，IPSへの全ての問い合わせに対して無条件に$RC_i$のユーザ名を返す方法が考えられる．しかし，この方法ではどの計算機からの問い合わせに対しても同様にユーザ名を返してしまうため，ユーザ名が不正取得される危険性がある．そこで，本論文ではRでIPパケット中のSYNフラグ，RSTフラグ，FINフラグを監視してコネクションの状態を管理し，IPSからの問い合わせに対してコネクションの有無を応答するようにする．このような機能を持つルータは我々の知る範囲では市販されていないが，UNIX系のOSを搭載した計算機上でこのような機能を持つプログラムを実装することは容易である．これにより，たとえSPとは無関係の計算機からIPSに問い合わせがあった場合でも，その計算機との間に問い合わせられたコネクションが存在しないためIPSはエラーを返すことができ，ユーザ名が不正取得される危険性を回避することができる．

### 3.3 ユーザ認証の全体手順

図2の構成におけるユーザ認証の全体の手順を以下にまとめる．以下の説明では，リモートアクセス計算機$RC_i$のユーザ$U_i$がダイアルアップ接続を行い，クライアントプログラム$CP_i$(使用するポート番号を$PR_i$とする)を用いてサーバプログラムSPにアクセスしたとき，SPがIDENTプロトコルを使って$U_i$を取得し，サービスを行う場合を例に取る．

(1) $U_i$は$RC_i$をRASにダイアルアップ接続する．この際，RASはASと協力して$U_i$が正規ユーザであるかどうかを確認し，そうであれば$RC_i$にIPアドレス$AR_i$を割り当て，ASに$(U_i, AR_i)$を記録する．もし，$U_i$が正規ユーザでなければ，RASは$RC_i$にエラーを返し，これ以上の処理を行わない．

(2) $U_i$は$RC_i$でクライアントプログラム$CP_i$を起動する．$CP_i$はSPとの間にコネクション$C_i$を確立する．

(3) Rは$C_i$の確立を認識し，その両端のIPアドレス並びにポート番号を記録する．

(4) SPは$C_i$の相手側のIPアドレスとポート番号を調べ，それぞれ$AR_i$と$PR_i$を取得する．

(5) SPはIPアドレス$AR_i$，ポート番号113との間にコネクションを確立しようとする．Rはこのコネクションを用いる通信に関して，SPの通信相手がIPアドレス$AR_i$，ポート番号113からそれぞれAI，$PI_i$となるように変換して中継する．

(6) SPはIPSとの間でコネクションを確立した後にIPSにコネクション$C_i$の両端のポート番号を送る．

(7) IPSは受信したポート番号からIDENTプロトコルによる本来の問い合わせ先のIPアドレス$AR_i$を取得し，またSPとの間のコネクションの情報よりSPの動作している計算機のIPアドレスを得る．また，SPから$C_i$の両端のポート番号を得る．これにより，IPSは$C_i$を特定するための全ての情報を得る．

(8) IPS は $C_i$ の両端の IP アドレス並びにポート番号を R に渡し，$C_i$ の存在の有無を問い合わせる．R は現在確立されているコネクションの記録を調べ，$C_i$ の有無を IPS に返す．

(9) IPS は，$C_i$ の存在を確認できなかった場合には SP にその旨を伝え，処理を修了する．また，$C_i$ の存在を確認した場合には AS に $AR_i$ を渡し，$RC_i$ のユーザ名を問い合わせる．AS は問い合わせに対して $U_i$ を返す．

(10) IPS は SP に $U_i$ を返す．SP はクライアントプログラム $CP_i$ のユーザが $U_i$ であると確認し，処理を行う．

(11) SP は処理を完了し，コネクション C を解放する．R はこの解放を検出し，$C_i$ に関する記録を削除する．

この方法では，SP や $CP_i$ を修正する必要がないので，ユーザの負担は非常に小さい．

# 4 ユーザ認証システムの実装

## 4.1 試作システムの構成

これまでに述べた方式の有効性を確認するため，我々はユーザ認証システムの試作を行った．試作したシステムの構成を図3に示す．この図からわかるように，試作システムでは1台の計算機にリモートアクセスサーバ(pppd)，IDENT 代理サーバ(ident proxy)，ルータ(natd)の全ての機能を持たせている．

この計算機はいわゆる AT 互換機で，OS として FreeBSD-2.2.7 を搭載している．リモートアクセスサーバとしては IIJ-PPP を用いており，ユーザ名と IP アドレスの対応は IIJ-PPP 自身が wtmp ファイルに記録する．この構成では認証サーバは不要であるが，3.3の(9)におけるユーザ名の取得は，IDENT 代理サーバが last コマンドを用いて行うようにしている．ルータの機能は OS 自身が有しているが，コネクション管理機能並びにアドレス変換機能は natd プログラムを一部修正したものを用いて実現している．主な修正点は，アドレス変換を必要としない通信についてもコネクションを管理するようにした点と，コネクションの確立・解放をファイルに記録するようにした点である．3.3の(8)におけるコネクションの確認は，ファイルに残されたコネクションの確立・解放の記録を分析するプログラムを新たに作成し，このプログラムに担当させている．IDENT 代理サーバは自作した同一のプログラムを inetd により複数のポートで問い合わせを待つようにしている．

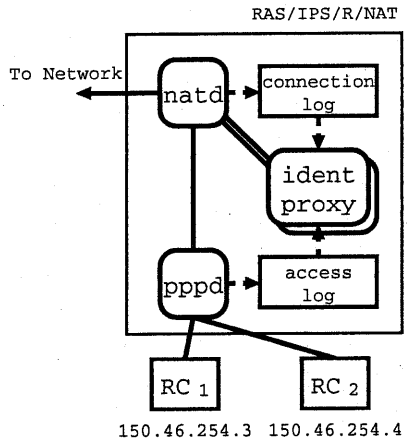


図 3: 試作システムの構成

## 4.2 運用結果

図3の構成において，我々は IDENT 代理サーバが正しく機能するかどうかを確認するため，試験運用を行った．以下ではその結果について述べる．

まず，IDENT プロトコルを用いてユーザ情報を得るプログラムとして sendmail を取り上げ，試験を行った．sendmail は標準で IDENT プロトコルによる発信者情報の取得を行い，ヘッダに付加する機能を有している．そこで試験では図3の構成において sendmail をサーバ計算機上で動作させ，リモートアクセス計算機からメッセージを送信した．その結果得られたヘッダの一部を図4に示す．この試験において，リモートアクセス計算機はユーザ kenya が利用しており，IP アドレスとして 150.46.254.3 が割り当てられている．ユーザ kenya はメッセージをサーバ計算機 ruin に発信する際，その From アドレスを 1 行めに示すように yamai@cc.okayama-u.ac.jp と偽ったが，2 行めに示すように Received ヘッダには (kenya@[150.46.254.3]) のようにリモートアクセス計算機のユーザ名と IP アドレスが記録されている．これにより，IDENT 代理サーバは正しく動作していることがわかる．

次に，IDENT プロトコルを用いてアクセス制御を行うプログラムとして tcpd を取り上げ，試験を

行った．tcpd はTCP による種々のサービスへの接続要求を監視し，アクセスを制御したり通信記録を取ったりするプログラムで，その機能の1つとしてIDENT プロトコルを用いてクライアントのユーザ名を求め，得られたユーザ名とクライアントのIP アドレスの組に基づいてアクセス制御を行う機能を有している．

試験では，サーバ計算機 ruin 上で tcpd を動作させ，hosts.allow ファイル，hosts.deny ファイルにそれぞれ `fingerd : kenya@150.46.254.3`, `fingerd : ALL` と記述してユーザ kenya からのアクセスだけを許可するように設定し，shima 並びに kenya の2通りのユーザ名でリモートアクセス計算機を接続して finger コマンドでサーバ計算機にアクセスした．その結果 tcpd が出力したログを図5に示す．これにより，IDENT 代理サーバは正しく動作し，tcpd がユーザ kenya からのアクセスだけを許すように機能していることがわかる．

以上の2つの例から，IDENT 代理サーバはリモートアクセス計算機のユーザを認証でき，IDENT プロトコルを用いる既存のプログラムと組み合わせて用いることができることが確認された．

## 5 まとめ

本論文では，リモートアクセス環境でもアプリケーションレベルで簡便にユーザ認証を行う方法として，IDENT 代理サーバを導入し，これにネットワーク接続時に認証したユーザ名を応答させる方法を提案した．また，この方法を実装し，既存のアプリケーションプログラムから IDENT プロトコルを用いてユーザ認証が行えることを運用実験により確認した．

この方法は既存のアプリケーションプログラムに何ら変更を加える必要がないため，多くのリモートアクセス環境に適用できると思われる．しかし，現在の構成では組織外で動作しているアプリケーションから見ると IDENT 代理サーバが信用できるかどうかわからないため，IDENT 代理サーバと同一組織内で動作しているサーバから認証を行う場合しか本方法は適用できない．今後の課題としては，他組織の IDENT 代理サーバが信頼できるかどうかを認証できる機構を導入し，上記の問題点を解決することが挙げられる．

```
From yamai@cc.okayama-u.ac.jp Fri Jan 22 22:09:01 1999
Received: from cc.okayama-u.ac.jp (kenya@[150.46.254.3])
        by ruin.cc.okayama-u.ac.jp (8.8.8/8.8.8) with
        ESMTP id WAA02023
        for <kenya@ruin.cc.okayama-u.ac.jp>; Fri, 22 Jan
        1999 22:11:52 +0900 (JST)
        (envelope-from yamai@cc.okayama-u.ac.jp)
```

図 4: sendmail における認証例

```
Feb  6 18:24:07 ruin fingerd[1044]: refused connect
 from shima@150.46.254.3
Feb  6 18:24:22 ruin fingerd[1045]: connect from
 kenya@150.46.254.3
```

図 5: tcpd における認証例

## 参考文献


[1] B. Lloyd and W. Simpson: "PPP Authentication Protocols", RFC 1334, IETF, October 1992.

[2] W. Simpson: "PPP Challenge Handshake Authentication Protocol (CHAP)", RFC 1994, IETF, August 1996.

[3] D. Atkins, W. Stallings and P. Zimmermann: "PGP Message Exchange Formats", RFC 1991, IETF, August 1996.

[4] Alan O. Freier, Phillip Karlton and Paul C. Kocher: "The SSL Protocol Version 3.0", draft-freier-ssl-version3-02.txt, Internet Draft, November 1996.

[5] M. St. Johns: "Identification Protocol", RFC 1413, IETF, February 1993.

[6] K. Egevang and P. Francis: "The IP Network Address Translator (NAT)", RFC 1631, IETF, May 1994.

[7] C. Rigney, A. Rubens, W. Simpson and S. Willens: "Remote Authentication Dial In User Service (RADIUS)", RFC 2138, IETF, April 1997.

[8] C. Rigney: "RADIUS Accounting", RFC 2139, IETF, April 1997.