静 岡 大 学　博 士 論 文

移動計算環境におけるユーザ認証に関する研究

１９９８年１月

大学院電子科学研究科

電子応用工学専攻

田 窪　昭 夫

博 士 論 文 目 次

専攻名・申請者　応用電子工学
氏　　名　田窪　昭夫

審査申請論文名　移動計算環境におけるユーザ認証に関する研究

目　　次

論文の要旨

ネットワーク、携帯型端末の技術進歩の結果、従来の計算機が固定設置された固定計算環境（ＦＣＥ）に加えて、計算機を携行して人の移動と共に計算環境が移動する移動計算環境（ＭＣＥ）が可能になってきた。ＦＣＥでは、計算機の設置と共にネットワーク自体も固定されるが、ＭＣＥでは無線を利用することにより、利用者と利用する場所を特定しない環境になっている。従ってＭＣＥでは新たな課題として、正当な利用者であることをネットワークに入る前に確認しなければならない。本論文は、この問題、すなわち、ＭＣＥにおけるユーザ認証に関して、ＭＣＥモデル、ユーザ認証プロトコル、プロトコルシミュレータを提案し考察した。

本論文は全６章から成る。第１章では本研究の背景、目的を述べた。第２章では本研究に関連する従来の研究動向を述べた。第３章ではＭＣＥにおけるユーザ認証の観点から、従来のＦＣＥの外挿としてＭＣＥのモデルを提案した。本方式では、移動端末を携行するモーバイルユーザは、最寄りのサーバに無線接続し、目的のホストに接続を託する。サーバがユーザ識

別子から簡単に接続先のホストを割り出せるようなユーザ識別子方式を提案した。第４章ではＭＣＥモデルの上で、ケルベロス方式を基本に電子パスポートを用いた新たなユーザ認証プロトコルを提案した。この方式では、第三者からの攻撃を考慮して、認証に関わる情報を少なく抑えられている。第５章では、外部からの第三者攻撃に対するユーザ認証プロトコルの安全性を確認するために、プロトコルシュミレータを作成し、ユーザ認証プロトコルの耐攻撃性の評価を概括した。第６章では本研究成果の応用分野への展開、今後の研究課題を述べた。

第1章　序論

1．1 研究の背景および目的

約半世紀前に誕生したコンピュータの利用形態は、当初はバッチ処理の利用形態が中心であり、特定の問題を解くために、計算機室において、問題を解くプログラムへデータ入力を行い、同じく計算機室において、その計算結果を得るという利用形態であった。やがて、1960年代には、ネットワークを経由して、論理的、物理的を問わず遠隔の端末から中央の処理装置を使用するオンライン処理がなされるような利用形態が実用になった。また、工業プラントや機械設備からの信号を、人手を介いさずにネットワークを経由して中央のコンピュータまで入力し、コンピュータによる自動計測制御処理を行うという利用形態も実用化された。

コンピュータの演算装置は初期には真空管が用いられていたが、その後トランジスタが用いられるようになり、さらにトランジズタを集積した集積回路（IC）が用いられるようになってきた。やがて、1970年代には、一つの集積回路の中にコンピュータの基本的な部分がすべて含まれたマイクロプロセッサが実現した。このマイクロプロセッサは初期には4ビット、または、8ビットの演算長を持つ小規模なものであったが、集積回路技術の急速な進歩に伴い、今日のパーソナルコンピュータに用いられているような32ビット演算長の高性能なマイクロプロセッサが実現してきた。

こうしたマイクロプロセッサの高機能・高性能化に合わせて、高集積化が推し進められるた結果、従来大型計算機、オフィスコンピュータと呼ばれ、専用の設置場所を余儀なくされていたコンピュータは、パーソナルコンピュータに代表されるように、デスクトップ・コンピュータ、ラップトップ・コンピュータ、ハンドヘルド・コンピュータと呼ばれるほどに、超小型・軽量なコンピュータが可能になってきた。これに伴い、企業に1台、あるいは、オフィスに1台であったコンピュータは、一人に1台のコンピュータと言われるほど広く普及した。

また、携帯・自動車電話が驚くべきほど増加し、一家に誰か一人は携帯電話を有してい

るような状況になってきた。また、昨年のWindows95に始まるパソコン市場の急速な発展、そして、インターネットの驚くべき普及、更にサブノートパソコン、ミニノートパソコン、PDA(Personal Digital Assistant)、HPC(ハンドヘルドパソコン)といった各種携帯情報機器も急速に発展し続けている。

一方、コンピュータ・ネットワークは、従来は中央のコンピュータと端末を結び付けるケーブルの延長という位置づけであったが、1969年米国でのARPANETにおけるパケット通信の成功により、単純なケーブルの提供という役割からパケットの分解、合成、伝送経路の設定などの各種の処理機能を持った分散情報処理の基盤としてのシステムとして発展してきた。この発展により、一層柔軟なコンピュータ間の接続が可能となってきた。また、ネットワーク利用の考え方も、特定企業・団体の利用に限定された利用の考え方に代わって、コンピュータ利用の共通基盤としての考え方が定着し、インターネットに代表される世界規模の開かれたネットワーク環境が整ってきた。

こうしたコンピュータ技術、コンピュータ・ネットワーク技術の進歩と共に、利用形態の多様化が目覚しく変わってきている。インターネット、携帯電話、PHS(Personal Handy-phone System)に代表される通信技術の発展と、サブノートパソコンに代表される携帯情報端末の発展により、移動しながら計算を行う環境が熟してきた。

コンピュータはネットワークを介して相互に接続され、データがリアルタイムで相互に行き来している。データ発生現場にコンピュータの端末が配置されている場合は、そこで発生するデータは、一瞬にしてネットワークに接続されたコンピュータでアクセスすることができる。 一方、コンピュータシステム利用の必要性は、端末の配置とは無関係にあらゆる場面で生じており、多くの場合データの発生現場には端末が配備されておらず、データ発生現場と端末の間には、物理的・時間的距離が存在した。

既存のコンピュータシステムは、予め決められた固定計算環境(フィックスドコンピューティング)、あるいは、半固定環境でデータの入力やアクセスを行っていた。しかしながら、

移動体で発生するデータを如何に迅速に、正確にコンピュータシステムへ取り込むか、また、こうした移動するデータ発生現場でのデータ処理・加工を可能にするかが課題となっている。

モーバイルコンピューティング（移動計算）の意義は、データ発生現場でのデータを如何に迅速、かつ、正確にコンピュータシステムへ取り込むことである。更に、現場に、オフィスと同じコンピュータ環境を移動させて持ち込み、データ発生現場でのデータ処理を可能にすることである。この結果、モーバイルコンピューティング環境（移動計算環境）とは、「いつでも、どこでも、誰とでも」をモットーにしたコンピュータ処理環境という事ができる。同時に、「誰とでも」の裏返しに、「誰からでも」ということから、インターネットのような開かれたネットワーク環境において、「誰とでも」、かつ、「誰からでも」、「安全に」通信できることが、モーバイルコンピューティング環境に求められる重要な課題である。

従来のコンピュータシステム環境では、End（端末）の設置場所が固定されており、コンピュータシステム利用のためには、わざわざそこまで足を運ばなければならない。 このようなコンピュータシステムを、FCE（fixed computing environment）と呼ぶのに対して、End（端末）の設置場所を離れて、どこからでもコンピュータシステムを利用出来る環境を、MCE（mobile computing environment）と呼ぶ。 また、FCEに関わるネットワークをFN（固定網）とよび、MCEに関わるネットワークを、MN（モーバイルネットワーク）と呼ぶ。

このようなモーバイルコンピューティング環境を構築しようとする際の課題は、

- ◆ 移動先から効率よく
- ◆ 移動先から安全にアクセスできること

本研究では、上記の課題を解決することを目的とし、モーバイルコンピューティング環境を構築する上で、特にセキュリティの観点から、以下の事項についての研究を行う。

(1) モーバイルコンピューティング環境のモデルの確立

(2) ユーザ認証プロトコルの設計

(3) ユーザ認証プロトコルの耐攻撃性の確認

1．2 論文の構成

本論文は全6章から成る。第1章では本研究の背景、目的を述べる。第2章では本研究に関連する従来の研究動向を述べる。第3章ではMCEにおけるユーザ認証の観点から、従来のFCEの外挿としてMCEのモデルを提案する。本方式では、移動端末を携行するモーバイルユーザは、最寄りのサーバに無線接続し、目的のホストに接続を託する。サーバがユーザ識別子から簡単に接続先のホストを割り出せるようなユーザ識別子方式を提案する。第4章ではMCEモデルの上で、ケルベロス方式を基本に電子パスポートを用いた新たなユーザ認証プロトコルを提案する。この方式では、第三者からの攻撃を考慮して、認証に関わる情報を少なく抑えられている。第5章では、外部からの第三者攻撃に対するユーザ認証プロトコルの安全性を確認するために、プロトコルシュミレータを作成し、ユーザ認証プロトコルの耐攻撃性の評価を概括する。第6章では本研究成果の応用分野への展開、今後の研究課題を述べる。

第2章　従来の研究の概観

2．1 モーバイルコンピューティングに関する従来の研究

従来はコンピュータの設置された場所でしか利用できなかったコンピュータ処理環境が、ネットワーク技術、とりわけ無線通信技術の進歩、および、コンピュータの小型化の結果、コンピュータそのものを持ち歩くことができるまでになり、コンピュータ処理環境を人の移動と共に、移動させることができるようになった。いわゆるモーバイルコンピューティング環境の実現である。

従来のコンピュータが固定設置された固定コンピュータ環境に対して、コンピュータそのものを移動させるモーバイルコンピューティング環境では、いろいろな可能性が生み出され、新たな試行がなされている。

コンピュータ処理の対象となるデータの発生は、コンピュータの設置場所とは無関係にあらゆる場面で発生している。従来の固定コンピューティング環境では、そうしたデータをその都度コンピュータの場所まで運ばなければ、コンピュータ処理することができなかったが、モーバイルコンピューティング環境では、コンピュータそのものをデータ発生現場に持ち込み、データ発生都度即座にコンピュータに取り込み、その場で処理することが可能になる。さらには、ネットワークを通じて、即座に中央のコンピュータに送り込んだりすることが可能になる。逆に、その場で必要なデータも、ネットワークを通じて、手元のコンピュータに取り込むことが可能になる。

こうしたモーバイルコンピューティング環境における新たな可能性については、[Imielinski-96] 、[水野-96]、[日経 BP-93]、[ACM-95]、などでいろいろ議論されている。また、モーバイルコンピューティング環境を実現するために欠かせない無線通信、ワイヤレス・コミュニケーションについては、 [Black-96]、[Brodsky-97]、[Gibson-96]、[Williams-96]、 [Dayem-97]に詳しく述べられている。モーバイルコンピューティング環境のこうした可能性を論理的に説明するために、モデル化の議論も行われている[Imielinski-96]。

2．2 セキュリティに関する従来の研究

ネットワークを通してコンピュータシステムを利用する場合、不正利用を防止するために、ユーザが正規のユーザであることが確認されなければならない。また、ユーザの利用状況は、第三者に漏れないように保護されなければならない。一般的に、前者は、ユーザ認証、後者は、プライバシ保護の問題として、いろいろと議論されてきている。

FCE環境におけるユーザ認証については、Kerberos などの例のように、いくつかの実験、実施例が見られる[RFC1510-93]。 また、MCE環境に似た携帯電話システムでも、ユーザ保護の観点からいくつかの実施例がみられる[Pfitzmann-97] [Samfat-94][Frankel-95]。

セキュリティの考え方、あるいは、定義などについては、Sinclaie、Cuppens 等が議論している[Sinclaie-96], [Cuppens-96]。 セキュリティのレベルを定性的に識別することについては、[Paulson-97a]（帰納的な方法によりセキュリティ特性を識別することを考察している。）、[Paulson-97b]（再帰的認証プロトコルの検証方式を考察している。）で議論された例がある。 プロトコルの観点から、セキュリティを定義したり[Roscoe-96]、セキュリティモデルを議論した例[Heintze-96]もある。

2．3 ユーザ認証に関する従来の研究

ユーザ認証の研究については、UNIX 分散環境での研究が古くからあり[Needham-78][Woo-92]、[RFC1004-87]、[RFC1507-93]でインターネット標準化されている。Schneider らは、CSP(Communication Sequential Processes)[Hoare-85][Milner-89]を用いて認証プロトコルの考察を行なっている。また、Lowe は、独自のプロトコル記述モデルから CSP 記述のプロトコルを生成する Casper コンパイラを作成している[Lowe-97a]。

Kerberos は、MIT で大学内のネットワークを構築するために、Needham と Schroeder

による認証方式[Steiner-88]をもとに、1983 年に IBM、DEC と共同で始めた Athena プロジェクトにおいて開発されたユーザ認証プロトコルである。Kerberos は、暗号化されたパスワードを利用するのではなく、第三者の認証センターが発行したチケットをやりとりし、パスワード自体が危険にさらされないようにしている。その後、Miller らにより改良が加えられ[Miller-87][Miller-88]、現在では、[RFC151093]でバージョン 5 がインターネット標準化されている[Kohl-93][Kohl-94][Neuman-94]。なお、Kerberos 方式を利用した認証方式として、ワンタイムパスワードとの組み合わせ方式がある[Clifford-95][Neuman-95b]。

ユーザ認証の必要性・意義については、[CCITT-88]、[Otway-87]、[Ragget-95]、[Burrows-90]、[Brown-95]、[Wilkers-95]などに詳しい。 ユーザ認証そのものの意義・定義については、[Gollmann-96]、[Zhou-96]などで議論されている。 Abadi 等[Abadi-96]は、プロトコル設計に関する原則を次のようにまとめている。

原則1:メッセージは、言語で(出来れば)一定の書式に基づいて記述され、かつ、意味のある内容でなければならない。

原則2:メッセージに対して施されるべき処置が明確になっており、受信者がメッセージの受け取り判断が可能になっていなければならない。

原則3:メッセージの内容理解に欠かせないキーワードは、適切に明示されなければならない。

原則4:セキュリティにつながらない冗長な暗号化は避けるべきである。

原則5:暗号化されたメッセージについて、プライバシの目的で暗号化された場合を除いて、発信者がその内容を既知であると推測すべきでない。

原則6:ノンス(nonce)については、それを後々どのように処理するかが、予め明確になっていなければらない。

原則7:カウンタなど予測性のデータは最新性の確認に有効な手段であるが、リプレイ攻撃に備えて、防御策を講じておかなければならない。

原則8:最新性の確認のためタイムスタンプを利用する場合は、当該メッセージに関わる計算機の時刻設定の差を、メッセージの有効時間より短い時間差に設定しておかなければならない。

原則9:一度使われた暗号鍵は、「古い鍵」である。

原則10:メッセージの内容理解に欠かせない暗号については、予めいずれの暗号方式であるかが明確になっていなければならない。

原則11:プロトコルが依って立つ信頼関係は何か、また、その理由・根拠が明確になっていなければならない。

ユーザ認証プロトコルについては、[Gong-95]、[Mao-95]、[Woo-94]、[Burrows-89]の研究がある。また、[Reiter-97]は ユーザ認証プロトコルの評価指標を試みている。 CSP を利用してユーザ認証プロトコルの記述も試みられている[Lowe-97b]。 インターネットにおける安全性については、[Lu-89]、[RFC1704-94]にまとめられている。

セキュリティの強さ・度合いなどの観点から、Focardi などの「セキュリティチェッカ」[Focardi-95][Focardi-96]の研究がある。 Focardi 等は、情報プロセスに伴うセキュリティ特性を記述する方法として、Milner の CCS[Milner-89]を拡張して、SPA(Security Process Algebra)を考案している。SPA では、CCS でのプロセスの行動について2段階の秘匿性 NNI(Non-deterministic Non Interference), NDC(Non Deductibility on Compositions)に加えて、情報プロセスのセキュリティ特性として、SNNI(Strong NNI), BNNI(Bisimulation NNI), BSNNI などを追加している。CCS でのプロセスの等価性をチェックするツールである CW(Concurrency Workbench)を改良して、SPA で記述された情報プロセスのセキュリティ特性の正当性を照査するツールとしてセキュリティ・チェッカ SC を考案している。SPA では、プロセス E をサブプロセスとする2つのプロセス C と D がそれぞれセキュリティ特性 X を持ち、かつ、等価であるとき、プロセス E もセキュリティ特性 X を持つと定義している。このことから、セキュリティ・チェッカ SC は、セキュリティ特性 X を持ち、かつ、情報プロセス

E をサブプロセスとする、SPA で記述された2つの情報プロセスを入力とし、それら2つの情報プロセスを解析して、それらの等価性を判断することにより、情報プロセス E のセキュリティ特性 X を判断する。

外部・第三者からの攻撃についての研究には、Aura による暗号を用いたプロトコルに対するリプレイ攻撃を回避する方法の考察[Aura-97] 、Schuba 等による TCP プロトコルにおける耐攻撃性の研究[Schuba-97]、Patel 等による RSA 暗号方式による鍵交換プロトコルに対する攻撃について、数論的な研究[Patel-97]、Lowe 等によるセキュリティプロトコルの弱点:新たな攻撃に関する研究[Lowe-96]、Bird 等による耐攻撃性を考慮した認証プロトコルの設計方式の研究[Bird-92]がある。 しかしながらモーバイルユーザの認証については、わずかに Molva の研究のみである[Molva-94]。

第3章　モーバイルコンピューティング環境モデル

3. 1 緒言

固定計算機環境(FCE)では、計算機は常時ネットワークの管理下にあり、ユーザは登録されている計算機のある場所に出向いて計算機ネットワークを利用する。また、そのサービスはそのネットワークに登録されているユーザのみが利用できる。

一方、モーバイルコンピューティング環境(MCE)では、ユーザは計算機を持って移動する。このような環境においてユーザがネットワークリソースを利用しようとする場合、ユーザは任意の移動先から登録先のリソースを利用できることが望ましい。

このようなモーバイルコンピューティング環境においてモーバイルユーザが移動した先でネットワークに接続する場合のパターンは2通り考えられる。

(1)登録されているネットワークへ接続する。

(2)手近なネットワークに接続し、そこを経由して自分が登録されたネットワークへ接続する。

前者の場合、ユーザの情報は接続先のネットワークに登録されている。そのため、接続要求を受け持ったサーバがユーザの身元を確認する場合、ユーザ確認のための情報、例えばユーザ ID やパスワードなどを直接参照できる。したがって、ユーザ認証を行う場合、通常のユーザ認証と同様の手法が利用できる。

一方、後者の場合、接続を受けるネットワークには、ユーザや移動計算機の情報は登録されていない。従って、接続を行ってきたユーザが誰なのか確認するために、ユーザ情報に直接アクセスすることができない。

この場合、認証を行うために必要な情報、例えばパスワードなどを、登録されているネットワークからユーザが接続を行ったネットワークに転送してしまうような方法も考えられる。しかし、これらの情報は非常に機密性の高い情報であり、別のネットワークにそのままの情報を渡してしまうのは避けるべきである。

そこで、モーバイルコンピューティング環境においてパスワードなどの機密性の高い情報を接続先のネットワークには転送せず、ユーザと接続先のネットワーク、ユーザが登録されているネットワーク間でなんらかの情報をやりとりし、ユーザの認証を行うプロトコルが必要になる。

モーバイルコンピューテイング環境においては、移動計算機からネットワークへのアクセスには、移動性などの点から携帯電話のような無線通信が使われることも考えなくてはならない。この無線通信をネットワーク接続に用いる場合、通常の有線ネットワークと比べ、

(1)低速でかつ低品質である

(2)接続コストが高い

(3)盗聴されやすい

などという点が問題になる。この点から、直接接続と間接接続を比較すると、常に直接接続を行うではなく、手近のネットワークに接続する間接接続の場合、無線ネットワークの利用を削減できるというメリットがある。

3.2 固定計算環境(FCE)と移動計算環境(MCE)

FCE(図3.1)での利用形態(表3.1)は、サーバへログインした後、利用環境が設定されるのが通常であり、End(端末)の設置場所へ赴き、End(端末)から、登録サーバへ、登録したユーザIDとパスワードでログインすることから始まる(図3.2)。 登録されていないサーバへはログイン出来ないのが一般的である。

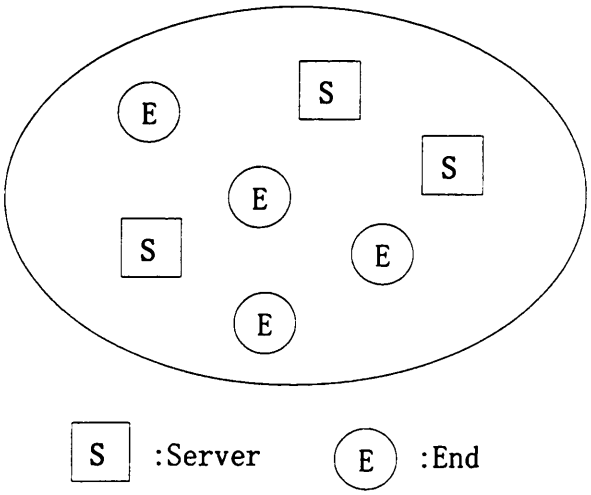

図3.1 FCE 環境

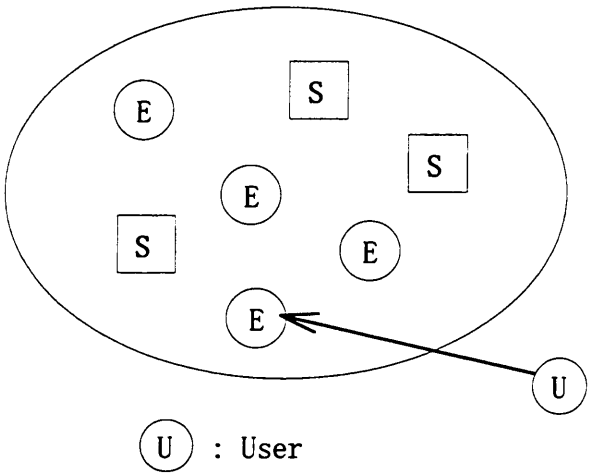

図3.2 FCE 環境の利用

表3.1　FCEでの利用形態

(1) 登録サーバへ、登録IDでログインする

非登録サーバへのログインはできない

所定のサービスが受けられる

他のサーバへは、当該登録IDで再ログイン(リモートログイン)出来る

GUESTで、他のサーバへ再ログインも出来る

(2) GUESTで、任意のサーバへログインする

サービスは限定される

他のサーバへの再ログイン(リモートログイン)は出来ない

MCE環境では(表3.2)、コンピュータシステム利用者はEnd(端末)の設置場所を離れる場合、たとえば、携帯電話接続機能を装備したノートパソコンを持ち歩く(図3.3)。 このことから、MCE 環境は、図3.4 のように、FCE 環境の外延として一般化される。 End(端末)の設置場所を離れた任意の場所から、ネットワーク接続の必要性が発生する都度、登録サーバへ無線接続してログインする。 この場合、効率の観点から、論理的、あるいは、物理的を問わず、遠く離れた場所から登録サーバへ無線接続してログインするのではなく、登録サーバへの接続要求を、最寄りのサーバ(非登録サーバ)で受け付けてもらい、そこを経由して登録サーバへログイン出来れば、都合が良い(図3.5)。

表3.2 MCEでの利用形態

(1) 登録サーバへ、登録IDでログインする

非登録サーバへのログインはできない

所定のサービスが受けられる

他のサーバへは、当該登録IDで再ログイン(リモートログイン)出来る

GUESTで、他のサーバへ再ログインも出来る

(2) GUESTで、任意のサーバへログインする

サービスは限定される

他のサーバへの再ログイン(リモートログイン)は出来ない

(3) 非登録サーバへ、登録サーバ、登録IDを名乗ってログインする

登録サーバ、登録ID保証の範囲で、サービスを受けられる。

(3.1) 非登録サーバを中継して、登録サーバへログインする(中継サービス)

(3.2) 登録サーバ、登録IDを保証として、ログインサーバのサービスを受ける

(1)のリモートログインの逆

(3.3) 登録サーバ、登録IDを保証して、他のサーバへログインする

登録サーバからログインすべきところを、ショートカットしてログインする

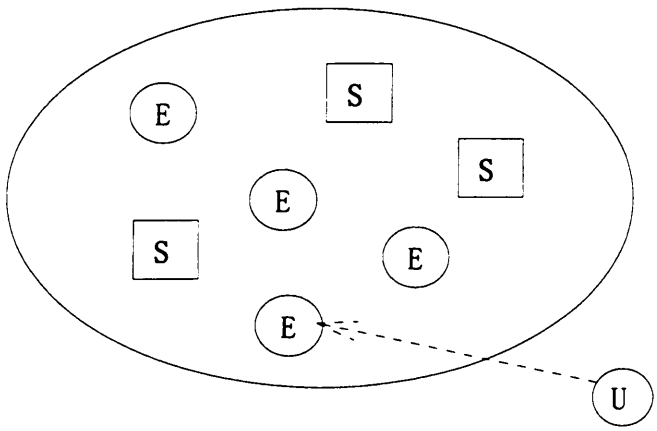

図3.3 モーバイルユーザの計算環境利用

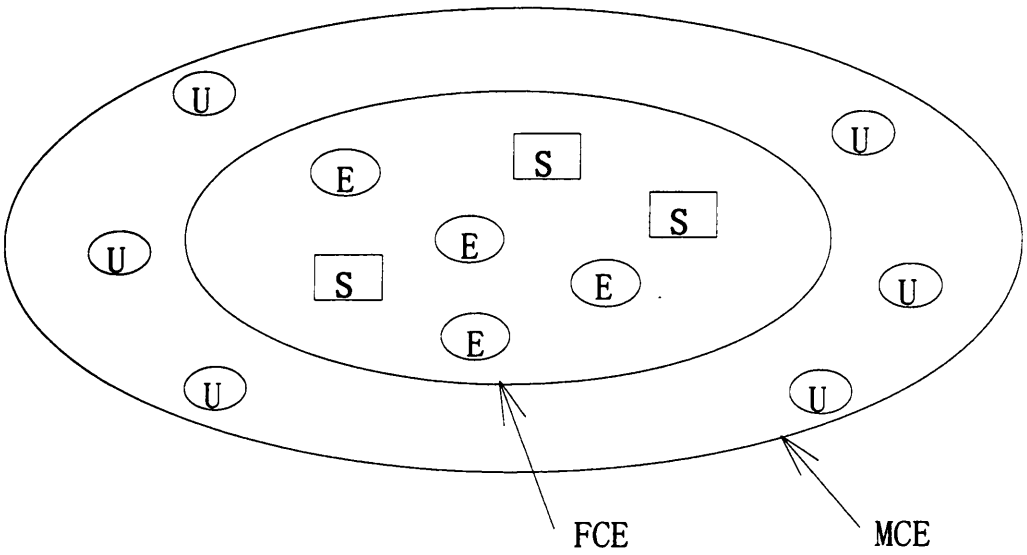

図3．4　MCE環境モデル

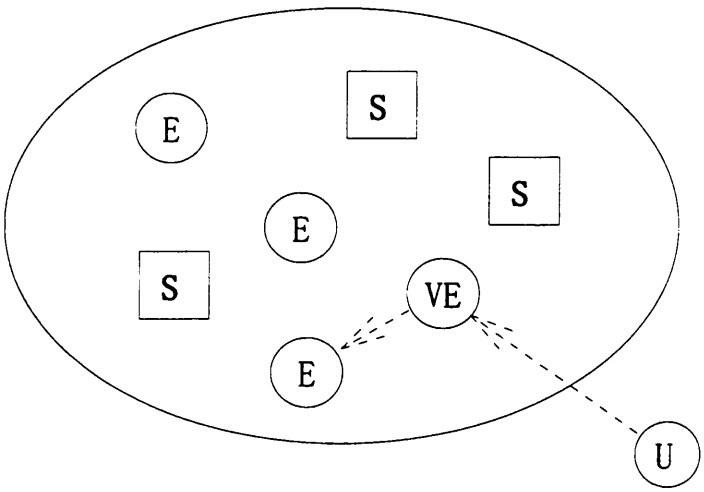

図3.5　遠隔モーバイルユーザの計算環境利用

この場合、単に(登録サーバへの)登録IDだけではなく、登録サーバも指定して無線接続することが考えられる。 こうすることにより、無線接続されたサーバは、指定された登録サーバへ問い合わせて、登録IDの有効性を確認して、許可されれば、ログイン操作を継続する。 登録サーバへの問い合わせでは、登録IDの有効性確認と共に、当該パスワードが返されるようにしておくことも考えられる。 登録IDの有効性の確認により、当該ユーザの身元保証が得られると、通常のログインの場合と同じように、引き続きパスワード入力をプロンプトして、正しいパスワード入力で、正常ログイン完了で、始めてネットワーク環境に入ることが出来る。このように、MCE環境では、MCE環境下に入る(ログイン)時点で、当該ユーザの身元保証、あるいは、ユーザ認証を行う必要がある。

端末からサーバにアクセスする環境では、ユーザが正当なアクセス権限の保持者であることを確認するため、ユーザ・リストがサーバに保持されている。 端末からのアクセス毎に、端末で入力されたユーザID、パスワードが正しいかどうかを、ユーザ・リストと照合して、正しければアクセスを許可し、正しくなければアクセスを拒否する。

複数のサーバが通信回線などで互いに接続されたコンピュータ・ネットワークでは、サーバ毎に、当該サーバのユーザ・リストを保持しておく。 ユーザは、コンピュータ・ネットワークにアクセスする場合、自分が登録されているサーバに接続する。 接続が許可された後、接続先のサーバを経由して、他のサーバへアクセスする。

この方式では、端末(ユーザ)の所在には無関係に、必ず自分が登録されたサーバに接続し(ログイン)、コンピュータ・ネットワークに入らなければならない。 端末の位置が固定されている従来のコンピュータ環境(FCE)で、一般的に採られている方式である。

一方、端末(ユーザ)の位置が固定されておらず、絶えず移動して、その位置を変えているモーバイルコンピューティングタ環境(MCE)では、必ず登録先のサーバを経由しないことには、コンピュータ・ネットワークに入り込めないのでは、接続時間、セキュリティなどの観点で最適な方法とは言いがたい。 たとえば、端末位置に最も近いサーバからネットーワ

ークに入れるようになれば都合が良い。

この場合、利用者の移動を常時監視して、移動先を常に把握しておくことも考えられるが、現実的な方法とは言いがたい。 サーバ側からは、意図的に利用者の移動を監視するのではなく、利用者からのアクセスがあった時にのみ、逐次利用者確認をして、アクセスの許認可を判断するのが、現実的な方法である。

また、利用者確認の観点から、サーバ毎に、すべてのサーバの利用者リストを保持することも考えられるが、運用の観点からは、適切な方法とは言いがたい。 また、いちいち全てのサーバへ利用者確認の照会をしていたのでは効率が悪い。 あるいは、ユーザIDと一緒に、登録サーバを指定したログインの方法も考えられるが、本論文では、FCE、MCEいずれの環境も区別することなく、シームレスな操作環境を維持するために、ユーザIDだけでのログインを考える。

ここで、ユーザID(とパスワード)の付与に工夫を凝らすことを考える。 IDは、ユーザだけに付与されるものではなく、サーバにもIDが付与されている。 これらのIDは、たとえば、IPアドレスの例に見るように、いずれも同じ体系にあるものとする。 サーバIDに変換を施して、ユーザID(とパスワード)を生成する。 また、この変換では、互いに異なる複数のユーザID(とパスワード)も生成出来る。 逆に、いずれのユーザID(とパスワード)にも逆変換を施せば、一意にサーバIDが引き出せる。

こうしたユーザID(とパスワード)の付与方式では、それぞれのサーバは、ネットワーク接続されたすべてのサーバID(たとえば、IPアドレス)だけを保持しておくだけで、移動ユーザからのアクセスに対して、常に登録先サーバが特定・確認でき、ユーザの正当性確認を迅速に行うことが出来る。

なお、前述のFCE、MCEは、ネットワークへの接続性からの区分である。一方、ネットワークの規模からの区分として、地球規模でのグローバル・ネットワークと、企業、大学など組織レベルでのローカル・ネットワークの区分が考えられる。ここでは、ローカル・ネットワ

ークに焦点を合わせたFCE、MCEについて考えることにする。

一般的に、AとBの通信においては、まずユーザAが主導者になって、ユーザAからサーバBにコネクション設定行う場合を考える。OSI7層モデルにおいて、トランスポート以下の低位の層と、セッション以上の層で対応に分けられる。IPv6は前者の代表である。また、上位層での対応については、Aから見て、接続先が、目的のBであることを確認するという意味から、「サーバ認証」と呼ばれている。一方反対に繋げられた先のサーバBにしてみれば、発信元が、ユーザAであることを確認するという意味から、「ユーザ認証」と呼ばれている。

ユーザ主導の観点では、まずユーザがネットワークに入る時点において、ユーザの正当性を保証するために、「ユーザ認証」が行われなければならない。「サーバ認証」は、ユーザが「ユーザ認証」が行われた後、接続されたサーバが正当なサーバであることをユーザに保証するために、「サーバ認証」が行われる。本論文では、モーバイルユーザが、ユーザ主導の立場でネットワークに入る時点での「ユーザ認証」についての研究である。Kerberosは、代表的な「ユーザ認証」プロトコルである。他に、PAP、CHAP[RFC1334-92]などの例がある。

インターネットで見られるように、ユーザAからサーバBへの接続に当たっては、一般的に、サーザBに行き着くまでに、いくつかのサーバ(ノード)を経て、目的のサーバBに接続される。所謂ホッピングを経て接続される。通常kerberosは、ユーザAがダイレクトにサーバBに接続する場合(シングルホッピング)のプロトコルであり、いくつかのサーバ(ノード)をホッピング(マルチホッピング)して、目的のサーザBに接続される場合は、途中途中のサーザ間で、発信元(ユーザ)の認証が行われる。

以下のモーバイルコンピューティングモデルでは、モーバイルユーザから、最寄りのFCEサーバまでは、シングルホッピング接続され、最寄りのFCEサーバから、目的のサーバまでは、一般的にマルチホッピング接続されると考える。

3. 3 ユーザIDとサーバID

ユーザID、サーバIDは、それぞれ、nビットの数値で表され、これを、n次の多項式で、次のように表現する。

$$C(x) = a_{n-1} \times x^{(n-1)} + a_{n-2} \times x^{(n-2)} + \cdots + a_1 \times x + a_0 \qquad (31)$$

$a_i = 0$ or 1 （i = 0, 1, 2, ... n-1）

ここで、生成多項式 $T(x)$を用いて、次の操作で、新たなIDを生成する。

$$T(x) \times C(x) \mod x^n - 1 \qquad (32)$$

$C(x)$に$T(x)$を乗じた結果を、$x^n - 1$ で除算した結果の剰余を、新たなIDとする。 サーバIDを、$C_0(x)$ として、次のように、ユーザID $C_1(x)$、$C_2(x)$ 、$C_3(x)$、.... を生成する。 除算結果の剰余を新たなIDとすることから、次のように、k番目の$C_k(x)$は、元の$C_0(x)$ に戻る。 このkを、当該$C_0(x)$ で生成されるID群（グループ）のエントリ数と呼ぶ。

$$\begin{aligned}
C_1(x) &= T(x) * C_0(x) \mod x^n - 1 \\
C_2(x) &= T(x) * C_1(x) \mod x^n - 1 \qquad (3.3) \\
C_3(x) &= T(x) * C_2(x) \mod x^n - 1 \\
C_4(x) &= T(x) * C_3(x) \mod x^n - 1 \\
&\cdot \\
&\cdot \\
C_k(x) &= C_0(x)
\end{aligned}$$

ここで、n = 6 , $T(x)$= x , $C_0(x) = x^3 + x^2 + 1$ の場合の例を、以下に示す。 この場合、当該IDグループのエントリ数は、6 となる。

$C_0(x) = x^3 + x^2 + 1$ ---------- 001101

$C_1(x) = x^4 + x^3 + x$ ---------- 011010

$C_2(x) = x^5 + x^4 + x^2$ ---------- 110100

$C_3(x) = x^5 + x^3 + 1$ ---------- 101001 (34)

$C_4(x) = x^4 + x + 1$ ---------- 010011

$C_5(x) = x^5 + x^2 + x$ ---------- 100110

$C_6(x) = x^3 + x^2 + 1 = C_0(x)$ ---------- 001101

$T(x) = x$ は、上の例から分かるように、$C_i(x)$ のビット列を、サイクリックに左シフトすることを意味する。この例では、たとえば、サーバIDを、$C_0(x)$=001101 に設定した場合、当該サーバのユーザIDを、$C_1(x)$、$C_2(x)$、$C_3(x)$、$C_4(x)$、$C_5(x)$ の中から選択することにより、ユーザID単独で、サーバIDを引き出すことが可能になる。

n＝6の場合（表3.3）、サーバID、ユーザIDのグループは、6エントリのグループが9(=k_6)グループ、3エントリのグループが2(=k_3)グループ、2エントリのグループが1(=k_2)グループ、それぞれ設定出来ることがわかる。また、n＝7の場合（表3.4）は、7エントリのグループが18(=k_7)グループ設定できる。

たとえば、n=6 の場合、$n = 6 = 1\times6 = 2\times3 = 3\times2 = 6\times1$ から、$2^6 = K_1 + K_2 + K_3 + K_6 = 1\times k_1 + 2\times k_2 + 3\times k_3 + 6\times k_6 = 1\times2 + 2\times1 + 3\times2 + 6\times9$ となる。nが素数の場合は、nエントリのグループが、$(2^n - 2)/n$ グループ設定出来ることが分かる。一般的に任意の数nについては、その因数に対応したエントリ数のグループに分かれる。また、ここでは、nビット全てが1のビット列、あるいは、0のビット列は、無意味なので、対象から除外する。

表3.5に、各nに対応したエントリ数、グループ数とビットパターンの組み合わせをまとめた。 表3.6に、各nに対応したエントリ数とグループ数の組み合わせをまとめた。また、表3.7には、各nと、その因数、および、Knを示した。

一般的に、任意の数nについて、1も因数のひとつに数え、n = i×j と表される場合、nビットパターンのグループのひとつに、iビットパターンを、j個連結したビットパターンがある。

iビットパターンのグループ数を，k_i で表す。各グループのエントリ数は，i で，グループのエントリ総数 K_i は，$i \times k_i$ である。一方、nビットパターンの総数は，2^n であることから，次の式が成り立つことが分かる。

$$2^n = \sum_{n=(i,j)} K_i = \sum_{n=(i,j)} i \times k_i \qquad (3.5)$$

n = 1 の場合は，1ビットパターンであり，0と1の2グループ（各グループのエントリ数は、それぞれ1である）から，$k_1 = 2$ であることが分かる。エントリ数の総数 K_1 は，$1 * k_1 = 2$ となる。

$$k_1 = 2; K_1 = 1 \times k_1 = 2$$

n = 1 × n からは、1ビットパターンを、n個繰り返すビットパターンと言うことで、nビット全てが1のビットパターンと、0のビットパターンの2グループ（k_1=2）（エントリ数 K_1 は，1 × k_1 = 2 である）であることが理解される。

n = n × 1 からは、nビットパターンを、1個繰り返すビットパターンであり、こうしたビットパターンを非対称ビットパターンと呼ぶ。一方、j≠1に対応するビットパターンを，対称ビットパターンと呼ぶ。また，jを対称次数と呼ぶ。

nが素数の場合は，1，n以外の因数を持たないことから（n = 1 × n = n × 1），nビットパターンの総数は，$K_1 + K_n$ である。このことから，素数nについて，$2^n = K_1 + K_n = 2 + n \times k_n$ となり、$2^n - 2$ は，nで整除されことが分かる。また、基数2を一般化して、mとし、nとmが互い素である場合を想定すると、容易にフェルマーの小定理が証明されることが分かる(付録.2)。

表3.3 n＝6の場合のIDリスト(k6 = 9, k3 = 2, k2 = 1, k1=2)

000001	000011	000101	000111	001001	001011	001101	001111
000010	000110	001010	001110	010010	010110	011010	011110
000100	001100	010100	011100	100100	101100	110100	111100
001000	011000	101000	111000		011001	101001	111001
010000	110000	010001	110001		110010	010011	110011
100000	100001	100010	100011		100101	100110	100111

010101	010111	011011	011111	000000	111111
101010	101110	110110	111110		
	011101	101101	111101		
	111010		111011		
	110101		110111		
	101011		101111		

表3.4 n＝7の場合のIDリスト(k7 = 18, k1=2)

0000000 1111111

0000001 0000011 0000101 0000111 0001001 0001011 0001101 0001111 0010011

0000010 0000110 0001010 0001110 0010010 0010110 0011010 0011110 0100110

0000100 0001100 0010100 0011100 0100100 0101100 0110100 0111100 1001100

0001000 0011000 0101000 0111000 1001000 1011000 1101000 1111000 0011001

0010000 0110000 1010000 1110000 0010001 0110001 1010001 1110001 0110010

0100000 1100000 0100001 1100001 0100010 1100010 0100011 1100011 1100100

1000000 1000001 1000010 1000011 1000100 1000101 1000110 1000111 1001001

0010101 0010111 0011011 0011101 0011111 0101011 0101111 0110111 0111111

0101010 0101110 0110110 0111010 0111110 1010110 1011110 1101110 1111110

1010100 1011100 1101100 1110100 1111100 0101101 0111101 1011101 1111101

0101001 0111001 1011001 1101001 1111001 1011010 1111010 0111011 1111011

1010010 1110010 0110011 1010011 1110011 0110101 1110101 1110110 1110111

0100101 1100101 1100110 0100111 1100111 1101010 1101011 1101101 1101111

1001010 1001011 1001101 1001110 1001111 1010101 1010111 1011011 1011111

表3.5 n＝1～ 6の場合の非対称エントリ数K_i、グループ数k_iとビットパターン

$n=1$　0　　$2^1=K_1=1\times k_1$, $K_1=2, k_1=2$

1

$n=2$　00　　$2^2=K_1+K_2=1\times k_1+2\times k_2$, $K_2=2, k_2=1$

11

01

10

$n=3$　000　　$2^3=K_1+K_3=1\times k_1+3\times k_3$, $K_3=6, k_3=2$

111

001　011

010　110

100　101

$n=4$　0000　　$2^4=K_1+K_2+K_4$, $K_4=12=4\times k_4, k_4=3$

1111

0101

1010

0001　0011　0111

0010　0110　1110

0100　1100　1101

1000　1001　1011

n=5　00000　　$2^5=K_1+K_5=1\times k_1+5\times k_5$, $K_5=30, k_5=6$

11111

00001　00011　00101　00111　01001　01011

00010　00110　01010　01110　10010　10110

00100　01100　10100　11100　00101　01101

01000　11000　01001　11001　01010　11010

10000　10001　10010　10011　10100　10101

n=6 000000 $2^6 = K_1 + K_2 + K_3 + K_6, K_6 = 54 = 6 \times k_6,\ k_6 = 9$

111111

010101 001001 011011

101010 010010 110110

100100 101101

000001	000011	000101	000111	001011	001111	010011	010111	011111
000010	000110	001010	001110	010110	011110	100110	101110	111110
000100	001100	010100	011100	101100	111100	001101	011101	111101
001000	011000	101000	111000	011001	111001	011010	111010	111011
010000	110000	010001	110001	110010	110011	110100	110101	110111
100000	100001	100010	100011	100101	100111	101001	101011	101111

表3.6 List of Entries/Groups

n	entries/groups : i / k_i	n	entries/groups : i / k_i
1	1/2	17	1/2,17/7710
2	1/2,2/1	18	1/2,2/1,3/2,6/9,9/56,18/14532
3	1/2,3/2	19	1/2,19/27594
4	1/2,2/1,4/3	20	1/2,2/1,4/3,5/6,10/99,20/52377
5	1/2,5/6	21	1/2,3/2,7/18,21/9858
6	1/2,2/1,3/2,6/9	22	1/2,2/1,11/186,22/190557
7	1/2,7/18	23	1/2,23/364722
8	1/2,2/1,4/3,8/30	24	1/2,2/1,3/2,4/3,6/9,8/30,12/335,24/698870
9	1/2,3/2,9/56	25	1/2,5/6,25/1342176
10	1/2,2/1,5/6,10/99	26	1/2,2/1,13/630,26/2580795
11	1/2,11/186	27	1/2,3/2,9/56,27/4971008
12	1/2,2/1,3/2,4/3,6/9,12/335	28	1/2,2/1,4/3,7/18,14/1161,28/9586395
13	1/2,13/630	29	1/2,29/18512790
14	1/2,2/1,7/18,14/1161	30	1/2,2/1,3/2,5/6,6/9,10/99,15/2182,30/35790267
15	1/2,3/2,5/6,15/2182	31	1/2,31/69273666
16	1/2,2/1,4/3,8/30,16/4080	32	1/2,2/1,4/3,8/30,16/4080,32/134215680

表3.7 List of K_n

n	factors of n	K_n
1	1	$K_1 = 2^1$
2	1,2	$K_2 = 2^2 - 2^1$
3	1,3	$K_3 = 2^3 - 2^1$
4	$1,2^2$	$K_4 = 2^4 - 2^2$
5	1,5	$K_5 = 2^5 - 2^1$
6	1,2,3	$K_6 = 2^6 - 2^3 - 2^2 + 2^1$
7	1,7	$K_7 = 2^7 - 2^1$
8	$1,2^3$	$K_8 = 2^8 - 2^4$
9	$1,3^2$	$K_9 = 2^9 - 2^3$
10	1,2,5	$K_{10} = 2^{10} - 2^5 - 2^2 + 2^1$
11	1,11	$K_{11} = 2^{11} - 2^1$
12	$1,2^2,3$	$K_{12} = 2^{12} - 2^6 - 2^4 + 2^2$
13	1,13	$K_{13} = 2^{13} - 2^1$
14	1,2,7	$K_{14} = 2^{14} - 2^7 - 2^2 + 2^1$
15	1,3,5	$K_{15} = 2^{15} - 2^5 - 2^3 + 2^1$
16	$1,2^4$	$K_{16} = 2^{16} - 2^8$
17	1,17	$K_{17} = 2^{17} - 2^1$
18	$1,2,3^2$	$K_{18} = 2^{18} - 2^9 - 2^6 + 2^3$
19	1,19	$K_{19} = 2^{19} - 2^1$
20	$1,2^2,5$	$K_{20} = 2^{20} - 2^{10} - 2^4 + 2^2$
21	1,3,7	$K_{21} = 2^{21} - 2^7 - 2^3 + 2^1$
22	1,2,11	$K_{22} = 2^{22} - 2^{11} - 2^2 + 2^1$
23	1,23	$K_{23} = 2^{23} - 2^1$
24	$1,2^3,3$	$K_{24} = 2^{24} - 2^{12} - 2^8 + 2^4$
25	$1,5^2$	$K_{25} = 2^{25} - 2^5$
26	1,2,13	$K_{26} = 2^{26} - 2^{13} - 2^2 + 2^1$
27	$1,3^3$	$K_{27} = 2^{27} - 2^9$
28	$1,2^2,7$	$K_{28} = 2^{28} - 2^{14} - 2^4 + 2^1$
29	1,29	$K_{29} = 2^{29} - 2^1$
30	1,2,3,5	$K_{30} = 2^{30} - 2^{15} - 2^{10} - 2^6 + 2^5 + 2^3 + 2^2 - 2^1$
31	1,31	$K_{31} = 2^{31} - 2^1$
32	$1,2^5$	$K_{32} = 2^{32} - 2^{16}$

3. 4 結言

モーバイルコンピューティング環境 (MCE) について、携帯情報端末、PDAなどを携行するモーバイルユーザが従来の固定コンピューティング環境にアクセスすることを想定したモーバイルコンピューティング環境モデルを考察し、MCE環境におけるユーザ認証の観点から、従来のFCE環境の外挿として、MCE環境のモデルを提案した。FCE環境の場合、利用者はネットワーク接続され、かつ、常時ホストに捕捉された端末に赴き、ログイン操作によりホスト計算機にアクセスする。一方、モーバイルユーザが携行する移動端末は、通常はネットワークに接続されておらず、必要に応じて逐次移動端末に備えられた無線機能を利用してネットワークに接続する。更に直接目的のホストに接続するのではなく、まず最寄りのサーバに無線接続し、自分が登録された目的のホストに接続を託する。この場合ユーザは登録先の目的ホストを指定することなく、単にユーザ識別子を最寄のサーバに伝えるだけで、最寄のサーバがユーザ識別子から簡単に接続先のホストを割り出せるような、ユーザ識別子方式を提案した。

本ユーザ識別方式では、ユーザ／サーバいずれの識別子も同じ体系で構成されており、ユーザIDから簡単な論理処理である巡回シフト (3.3式、および、$T(x) = x$) でサーバIDを一意に割り出すことが可能である。現在インターネットで広く利用されているビットマスク処理に比較して、巡回シフトは容易にハードウエア的に実装することが可能であるのも一つの特徴である。

第4章 ユーザ認証プロトコル

4. 1 緒言

世界的規模のモーバイルコンピューテイング環境においては、ユーザが任意の移動先から未登録ネットワークを経由し、登録済のリソースにアクセスできることが望ましい。そのためには、未登録ネットワークのアクセスポイントにおいて第三者認証を基本としたユーザ認証が不可欠である。この目的のために、本論文では、無線ネットワークの利用をなるべく少なくし、ユーザ情報を外部に洩らすこと無く、ユーザを確認できる広域ユーザ認証プロトコル GMAP(Global Mobile Authentication Protocol)を提案する。

4. 2 MCEにおけるユーザ認証プロトコル

GMAP では、モーバイルユーザの接続要求を受けてユーザの確認を行うモーバイルサーバを導入し、Kerberos を基本とした第三者認証を行う。

Kerberos は、もともと図4.1に示すように、ユーザがサーバにアクセスする際、サーバに対して、ユーザの身分を明らかにするための方法として、パスワードなどを利用するのではなく、図4.2に示すように、ユーザ、サーバのいずれとも無関係な第三者の認証局を導入する方式で、ユーザは、まず認証局にアクセスして、予め登録された情報に従って、身分を証明するための「チケット」の発行を申し出て入手する。ユーザはこのチケットをサーバに送りつける。

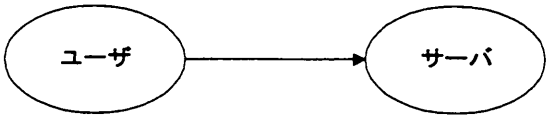

図4.1　当事者によるユーザ認証

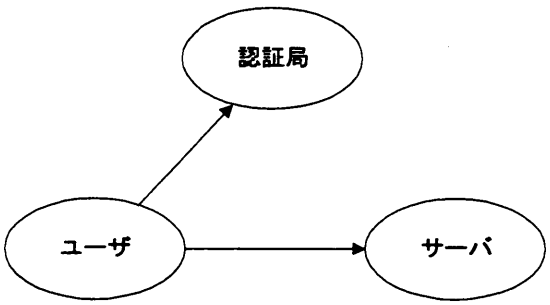

図4.2　第三者によるユーザ認証

通常ネットワーク上では、図4.3に示すように、目的のサーバに至るまでの間、いくつものサーバを中継する(マルチ・ホッピング)。 Kerberos 方式をそのまま適用すると、図4.4のように、中継サーバ毎に、サーバは認証局からチケットをもらい受け、次のサーバへの接続に利用する。

図4.3　マルチホップの場合の当事者によるユーザ認証

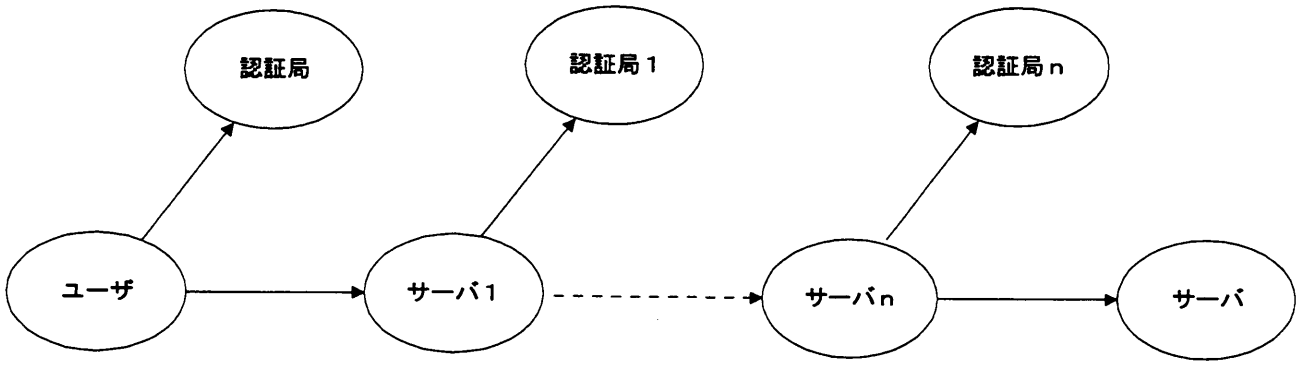

図4.4　マルチホップの場合の第三者によるユーザ認証

ここでは、前章で述べたモーバイルコンピューティング環境(MCE)について、図4.3において、ユーザをモーバイルユーザ(MU)と位置づけ、サーバ1をユーザに最も近いモーバイルサーバ(MS)と位置づける。サーバはモーバイルホスト(MH)と位置づける。サーバに至るまでの中継サーバｎは、以下議論では、省略できるので、図4.5について議論する。

図4.5　モーバイルユーザ認証モデル

未登録のネットワークに接続したモーバイルユーザ(MU)は、そのネットワークのモーバイルサーバ(MS)に対して自分の身元を明かさなければならない。この時、MS は MU の身元を自身で確認することができない。そこで MS は MU の登録されているモーバイルサーバ(MH)と通信を行い、その身元を確認することになる。

この時、ホストのなりすましなどを考えると、通信相手が目的の相手である事を確認することが必要になる。また、経路上でのデータの盗聴や改竄の危険性を考慮すると、重要な情報は通信相手にしかわからない形態、つまり暗号化などを行ってやりとりすることを考えなければならない。

そこで、Kerberos に見られるような第三者認証を行うための認証サーバ(AS)、チケットサーバ(TS)を導入する。 通常の Kerberos では、ユーザが AS, TS への問い合わせを行うが GMAP では、AS, TS と通信を MS が代行する形態をとる。これにより、無線通信の利用を削減でき、前述の無線通信の弱点を避ける事ができる。

MCE環境のセキュリティ・モデルとして、ユーザ認証システム Kerberos の適用を試みると、図4.6.認証モデルAのようになる。図で、MUはモーバイルユーザ、MHはモーバイルホスト(登録サーバ)、MSはモーバイルサーバ(非登録サーバ)、ASは認証サーバ、TSはチケットサーバを表す。 図4.6bは、一般形である図3.6aの中で、モーバイルサーバMSがモーバイルホストMHの場合である。図中、細線は無線接続(WL)を、また、太線は有線接続(CL)(FCE環境)を表す。

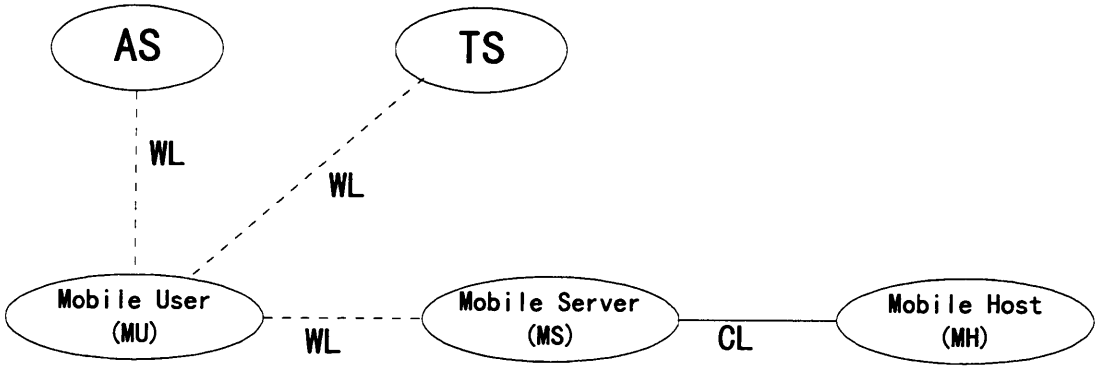

図4.6a　Authentication Model-A

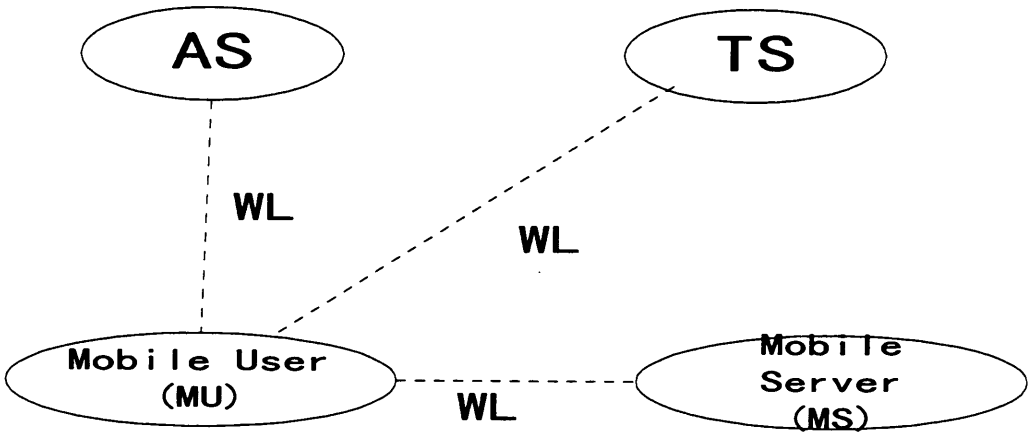

図4.6b　Authentication Model-A

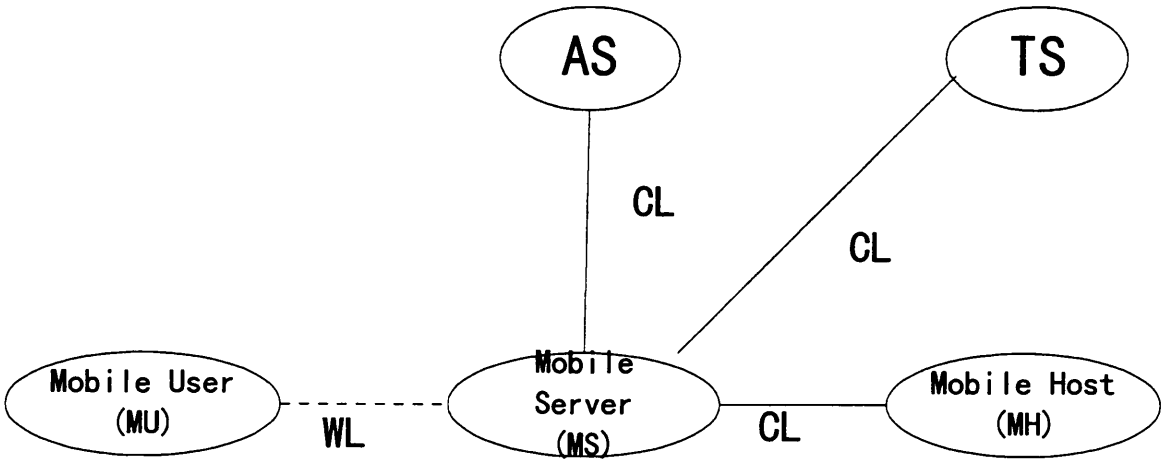

図4.7a　Authentication Model-B

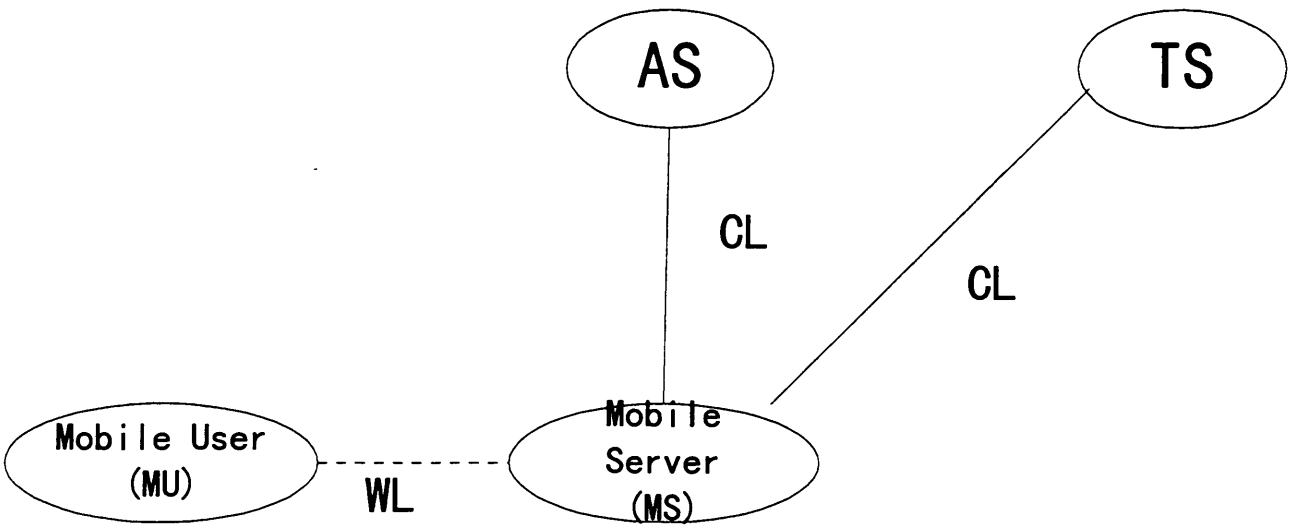

図4.7b　Authentication Model-B

図4.6.認証モデルAの場合、MUが無線を利用して、AS、TSと通信して、自分の身元の裏付けを取り付ける方式である。また、図3.6bの場合であっても、MUはAS、TSに身元保証を取り付けている。この身元保証を条件に、MS、または、MHを経由して、ネットーワークへのログイン手続きが行われる。場合によっては、MSの身元保証を取り付けることを考えてもよい。

一般的に、モーバイルユーザは、通常はネットワークに接続されていなくて、その必用性が発生する都度、近くのサーバを経由して、ネットワークに接続する(ログイン)ことから、図4.7a.の認証モデルBを提案する。有線接続(FCE)に比較して、無線接続は低速、かつ、接続費用がかかることから、図4.6.認証モデルAに比較して、無線接続による通信量・回線占有時間を最少限に押さえることが出来る。 図4.7b は、図4.6bに対応して、MSがMHの場合である。 図4.7.認証モデルBでは、MUは身元保証としてパスポートを保持して、ユーザIDと一緒に、MSに提示して、ネットワークへのログイン認可を申し出る。 MHは、MUから送られてくるパスポートだけで、MUの身元確認を行うことができる。一方、MSの誠意ある中継処理を保証するために、MSの身元保証を確認する。

図4.7.認証モデルBにおけるMUのユーザ認証の手順は次の通りである。説明中の①～⑧は、図4.8に対応している。ここで、MUは、自分のユーザID、秘密キー、および、パスポートを保持している。また、パスポートには、MUのユーザID、電子署名などが記述されている。ユーザIDは、前述の方式に準拠して生成されたものである。なお、MUのパスポートは、MUの秘密キー sKmu で暗号化され、かつ、MHの公開キー pKmh で二重の暗号化されるが、それぞれのキー生成に使用される素数組みの積の大小関係を設定することにより、二重に暗号化されたMUのパスポートは正常に復号化できる[Kohnfelder-78] 。

①MU ---> MS : IDmu,((Passport)sKmu)pKmh

MUは、ユーザIDと、MUの秘密キー sKmu で暗号化したパスポートを更にMHの公開キー pKmh で暗合化したものを、MSに送る。パスポートは、送信者がMU本人であることの証しに、MUの秘密キー sKmu で暗号化し、さらに、MH以外にその中身を見られないように、MHの公開キー pKmh で暗合化する。MSは、ユーザIDから、MHのIDを割り出して、自分がユーザの登録サーバであるか否かを判定する。 図4.7aはMSが非MHの場合であり、図 4.7bはMSがMHの場合である。 図4.7bの場合は、そのまま通常のログイン手続きへ進むことになる。 以下、図4.7aの場合の手順を述べる。

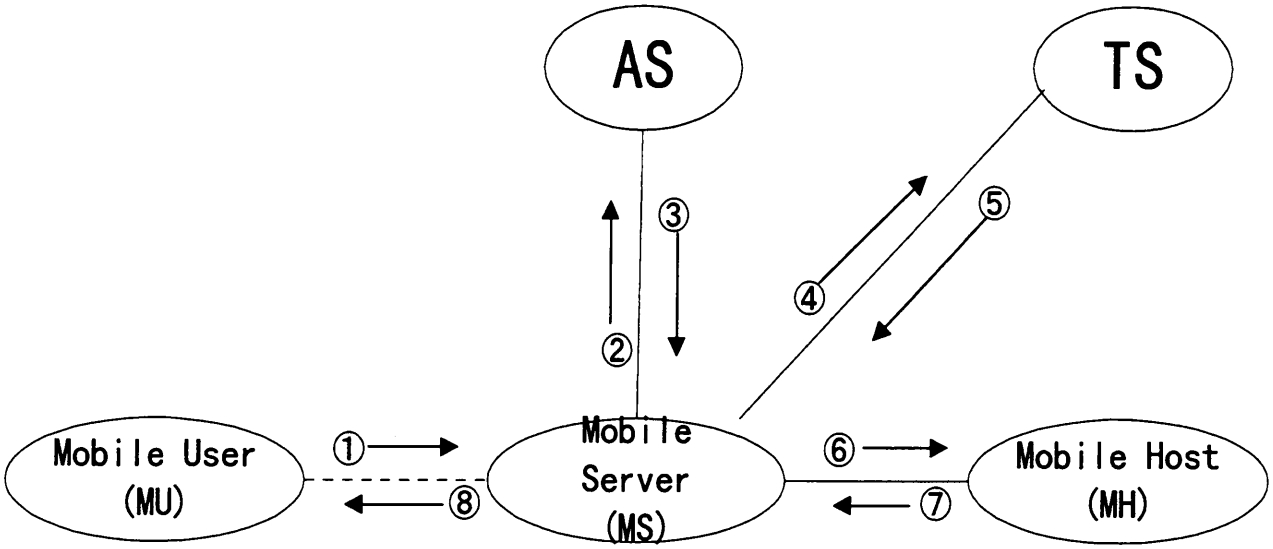

図4.8　認証モデルBのユーザ認証手順

②MS ---> AS : IDms,IDts

MSは、MSとTSのIDをASへ送る。

③AS ---> MS : (Kms,ts)pKms (Ticket-A)pKts

ASは、MSとTSの交信のためのセッションキーKms,ts、および、 Ticket-Aを作成する。TSの公開キー pKts で暗号化した Ticket-A と、 MSの公開キー pKms で暗号化したセッションキー Kms,ts を送り返す。 Ticket-A には、MS、TSのIDのほか、セッションキーKms,ts、発行時刻、有効期限が記されている。

④MS ---> TS : (IDms,CurrentTime)Kms,ts, (Ticket-A)pKts, IDmh

MSは、復号化して、セッションキー Kms,ts とTSの公開キー pKts で暗号化されたTicket-A を取り出す。MSのIDと現在時刻をセッションキー Kms,ts で暗合化して、一緒にTSの公開キー pKts で暗合化されたTicket-A とMHのIDを、TSに送る。

⑤TS ---> MS : (Kms,mh, (Ticket-B)pKmh)Kms,ts

TSは、復号化して、 Ticket-A を取り出す。有効期限が切れていないことを確認して、Ticket-A の中のセッションキー Kms,ts で復号化したMSのID、現在時刻が、 Ticket-Aの内容と一致することを確かめる。MHのIDを確認して、MSとMHの交信のためのセッションキー Kms,mh、および、 Ticket-B を作成する。 MHの公開キー pKmh で暗号化したTicket-B とセッションキー Kms,mh を、セッションキー Kms,ts で暗号化して、MSへ送り返す。 Ticket-B には、MS、MHのIDのほか、セッションキー Kms,mh、発行時刻、有効期限が記されている。

⑥MS ---> MH : (IDms, CurrentTime)Kms,mh, (Ticket-B)pKmh, ((Passport)sKmu)pKmh

MSは、復号化して、セッションキー Kms,mh、および、MHの公開キー pKmh で暗合化した Ticket-B を取り出す。MSのIDと現在時刻をセッションキー Kms,mh で暗号化して、一緒にMHの公開キー pKmh で暗合化された Ticket-B、MUの秘密キー sKmu とMHの公開キー pKmh で二重に暗合化されたMUのパスポートを、MHに送る。

⑦MH ---> MS : (ConfirmationMessage)Kms,mh

MHは、復号化して、 Ticket-B、MUの秘密キー sKmu とMHの公開キー pKmh で暗合化されたMUのパスポートを取り出す。有効期限が切れていないことを確認して、Ticket-B の中のセッションキー Kms,mh で復号化したMHのID、現在時刻が、 Ticket-Bの内容と一致することを確かめる。MUの公開キー pKmu で復号化したMUのパスポートから、MUの電子署名を確認して、正規のユーザであることを認知する。この認知情報を、セッションキー Kms,mh で暗号化して、MSへ送り返す。認知情報には、ユーザ判定結果とユーザ・パスワード(正当ユーザの場合、ログイン手続きに必要な)が記されている。

⑧MS ---> MU : LoginPromptMessage

MSは復号化して、認知情報を確認する。MUがMHの登録ユーザであることが確認されれば、通常のネットワークへのログイン手続きの処理へ進む。

同様に、図4.6.認証モデルAにおけるユーザ認証プロトコルは、次の通りである。説明中の①～⑧は、図4.9に対応している。

①MU ----> AS : IDmu, IDts

MUは、MUとTSのIDをASへ送る。

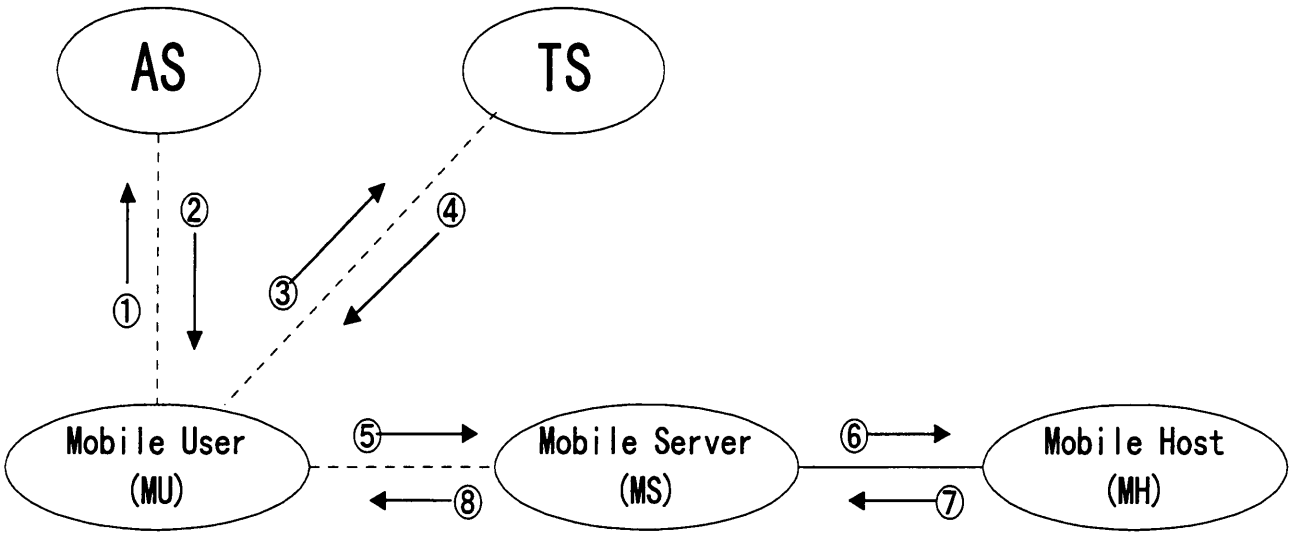

図4.9　認証モデルAのユーザ認証プロトコル手順

②AS ----> MU : (Kmu,ts)pKmu (Ticket-A)pKts

ASは、MUとTSの交信のためのセッションキー Kmu,ts、および、 Ticket-A を作成する。TSの公開キー pKts で暗号化した Ticket-A と、 MUの公開キー pKmu で暗号化したセッションキー Kmu,ts を送り返す。 Ticket-A には、MU、TSのIDのほか、セッションキー Kmu,ts、発行時刻、有効期限が記されている。

③MU ----> TS : (IDmu, TimeStamp)Kmu,ts, (Ticket-A)pKts,IDms

MUは、復号化して、セッションキー Kmu,ts と、TSの公開キー pKts で暗号化されたTicket-A を取り出す。 MUのIDと現在時刻をセッションキー Kmu,ts で暗合化して、一緒にTSの公開キー pKts で暗合化された Ticket-A と、MSのIDを、TSに送る。

④TS ----> MU : (Kmu,ms, (Ticket-B)pKmu)Kmu,ts

TSは、復号化して、Ticket-A を取り出す。有効期限が切れていないことを確認して、Ticket-A の中のセッションキー Kmu,ts で復号化したMUのID、現在時刻が、Ticket-Aの内容と一致することを確かめる。 MSのIDを確認して、MSとMUの交信のためのセッションキー Kmu,ms、および、 Ticket-B を作成する。 MUの公開キー pKmu で暗号化した Ticket-B とセッションキー Kmu,ms を、セッションキー Kmu,ts で暗号化して、MUへ送り返す。 Ticket-B には、MS、MUのIDのほか、セッションキー Kmu,ms、発行時刻、有効期限が記されている。

⑤MU ----> MS : (IDmu, TimeStamp)Kmu,ms, (Ticket-B)pKms,((Passport)sKmu)pKmh

MUは、ユーザIDと有効期限をセッションキー Kmu,ms で暗号化したもの、MSの公開キー pKms で暗号化した Ticket-B 、および、MUの秘密キー sKmu で暗号化したパスポートを更にMHの公開キー pKmh で暗合化したものを、MSに送る。 パスポートは、送信者がMU本人であることの証しに、MUの秘密キー sKmu で暗号化し、さらに、MH以外にその中身を見られないように、MHの公開キー pKmh で暗合化する。 MSは、ユーザIDから、MHのIDを割り出して、自分がユーザの登録サーバであるか否かを判定する。図4.6aはMSが非MHの場合であり、図 4.6.b はMSがMHの場合である。 図4.6b の場合は、そのまま通常のログイン手続きへ進むことになる。 以下、図4.6aの場合の手順を述べる。

⑥MS ----> MH : ((Passport)sKmu)pKmh

MSは、MUの秘密キー sKmu とMHの公開キー pKmh で暗合化されたMUのパスポートをMHへ先送りする。

⑦MH ----> MS : (ConfirmationMessage)pKms

MHは、MHの秘密キー sKmh、および、MUの公開キー pKmu で復号化したMUのパスポートから、MUの電子署名を確認して、正規のユーザであることを認知する。この認知情報を、MSの公開キー pKms で暗号化して、MSへ送り返す。認知情報には、ユーザ判定結果と(正当ユーザの場合、ログイン手続きに必要な)ユーザ・パスワードが記されている。

⑧MS ----> MU : LoginPromptMessage

MSは復号化して、認知情報を確認する。MUがMHの登録ユーザであることが確認されれば、通常のネットワークへのログイン手続きの処理へ進む。

4. 3 認証プロトコルの記述方式

前節で記述した認証プロトコルの記述方式について述べる。シンボルA、Bは、前述のMU、MS、MHなどの当事者を表す。シンボルSは、前述のAS、TSなどの第三者サーバを表す。 sKa、sKbは、それぞれA、Bのシークレット・キーを、また、pKa、pKbは、それぞれ A、Bの公開キーを表す。Ka,b は、A、B同士の間のセション・キーを表す。

AからBへメッセージ“message”を送ることを、次のように記述する。

A ----> B : <message> (4.1)

"message"を、キーsKaで暗号化したものを、(message)sKa と記述することとすれば、“message”は、一般的に、Backus-Naur 記法で、次のように記述される。

<message>::=<message>|<(message)sKa>|<(message)pKb>|<(message)Ka,b>

また、sKa、pKaなどの性質から、((message)sKa)pKa は、message そのものであることから、一般的に、次の諸式が設定される。

((message)sKa)pKa=message

((message)pKb)sKb=message

((message)Ka,b)Ka,b=message (4.2)

(message1+message2)pKa=(message1)pKa+(message2)pKa

(message1+message2)sKa=(message1)sKa+(message2)sKa

((message1+message2)pKa)sKa=((message1)pKa+(message2)pKa)sKa

=((message1)pKa)sKa+((message2)pKa)sKa

=message1+message2

((message1+message2)sKa)pKa=((message1)sKa+(message2)sKa)pKa

=((message1)sKa)pKa+((message2)sKa)pKa

=message1+message2

次に、上記記述方式によるユーザ認証プロトコルを、LIPS 処理系で解析できるように、表現(4.1)を、LISP に準じて、次のように記述する。

((A)(B))(message) (4.3)

表現(4.3)では、A、B が当事者を表し、message がメッセージを表していることが暗黙の前提になっているが、それぞれが、当事者であり、また、メッセージであることの明示するために、種別識別子を用いて、正確には次のように記述されものとする。

(((A)(AGENT))((B)(AGENT)))((message)(MESSAGE)) (4.4)

ここで、「AGENT」は、当事者を表す識別子であり、「MESSAGE」メッセージを表す識別子である。これらの識別子を含め、次に挙げるのような識別子を用意する。

AGENT	当事者(エージェント)
MESSAGE	メッセージ
SERVER	サーバ(第三者認証サーバなど)
UID	ユーザID
NONCE	意味のない情報

KEY	キー
SKEY	秘密キー
PKEY	公開キー
TICKET	チケット
PASSPORT	パスポート
TIME	時刻

また、メッセージの連記、例えば、message1 message2 message3は、混乱を招かなければ、次のいずれかで記述されてもよいことが分かる。

((message1)((message2)(message3))) (4.5)

((message1)(message2)(message3)) (4.6)

また、暗号化されたメッセージ、例えば、(message)sKaは、次のように記述される。

((message)(MESSAGE))((sKa)(SKEY)) (4.7)

あるいは、簡単に、次のように記述する。

(message)(sKa) (4.8)

(4.7)、(4.8)は、messageを秘密キー sKaで、暗号化することを記述したものであり、ユーザ認証プロトコルの記述には十分であるが、将来のユーザ認証プロトコル解析を想定して、メッセージの復号化は、キーを前置して、次のように記述することにする。

(pKa)(message) (4.9)

また、混乱を招かないのであれば、次の2つは、同じことを意味することになる。

(pKa)((message)(sKa)) (4.10)

((message)(sKa))(pKa) (4.11)

(4.10)、(4.11)は、正規法で記述すると、次の通りである。

((pKa)(PKEY))(((message)(MESSAGE))((sKa)(SKEY))) (4.12)

(((message)(MESSAGE))((sKa)(SKEY)))((pKa)(PKEY)) (4.13)

前述の認証モデルBで示したユーザ認証プロトコルを、上述の記述方式で書き直すと、次のようになる。

((MU)(MS))((IDmu)(((Passport)(sKmu))(pKmh)))

((MS)(AS))((IDms)(IDts))

((AS)(MS)) (((Kms,ts)(pKms))((Ticket-A)(pKts)))

((MS)(TS))((((IDms)(CurrentTime))(Kms,ts))((Ticket-A)(pKts))(Idmh))

((TS)(MS))(((Kms,mh)((Ticket-B)(pKmh)))(Kms,ts))

((MS)(MH))(((IDms)(CurrentTime))(Kms,mh))

((Ticket-B)(pKmh))(((Passport)(sKmu))(pKmh))

((MH)(MS))((ConfirmationMessage)(Kms,mh))

((MS)(MU))(LoginPromptMessage)

4．4 認証モデルAの経路上での攻撃

外部からの攻撃は、特定のMUを狙った攻撃と、ネットワークを狙った攻撃に分けられる。

特定のMUを狙った攻撃では、特定のMUに関係する情報だけを対象に、当該情報を盗んだり、改竄を行おうとする攻撃である。また、ネットワークを狙った攻撃は、ネットワーク上を流れる全ての情報を対象にして、それらを利用することにより、ネットワークへの不正な侵入を試みる攻撃である。

MUに関する情報に着目すると、認証モデルBでは、MU・MS間とMS・MH間だけで流れるのに対して、認証モデルAでは、全ての経路でMUの認証情報であることを識別できる情報が含まれている。このことから、認証モデルAと認証モデルBでは、MUを狙った

攻撃に対するセキュリティ強度の差があることが分かる。

以下、ネットワークを狙った攻撃について、認証モデルA、認証モデルBの各経路上での攻撃の可能性について検討する。

(a)MU・AS間の攻撃

MU・AS間を流れる情報は、(1)MUからASへのリクエストにあたるIDmuとIDts、および、(2)ASからMUへ返されるセションキー Kmu,ts をMUの公開キー pKmu で暗号化したものと、TSの公開キー pKts で暗号化したチケット Ticket-A である。

(1)に関しては、IDmu がそのまま流れるので、それが盗聴されると、当該ユーザがどのドメインにいるかが容易に判明されてしまう。IDts については、公開情報なので、盗聴されても問題にはならない。

2つの ID をリプレイしたとしても、MU になりすますことはできない。なぜならば、AS から送られてくる情報は MU の公開鍵で暗号化されているので、それを解読できない限り、それ以降の処理が続けられないからである。

(2)に関しては、AS から送り返される情報は、MU の公開キー pKmu、および、TSの公開キー pKts で暗号化されているので、もし盗聴されたとしても、意味のある情報は取り出すこともできなければ、意味のある改竄も不可能である。また、チケット Ticket-A にタイムスタンプを忍び込ませておくことにより、リプレイアタックをできなくすることが可能である。

(b)MU・TS間の攻撃

MU・TS間を流れる情報は、(1)MUからTSへ送られる(IDmu, TimeStamp)Kmu,ts と (Ticket-A)pKts、および、(2)TSからMUへ送られる(Kmu,ms, (Ticket-B)pKmu)Kmu,ts である。

(1)に関しては、セションキー Kmu,ts、および、TSの公開キー pKts で暗号化されている

ので、意味のある情報は取り出すこともできなければ、意味のある改竄も不可能である。また、これらの情報を使ってリプレイアタックしたとしても、TSから送られてくる情報がセションキー Kmu,ts で暗号化されており、それを解読できない限り、それ以降の処理が続けられないので、MUになりすますことはできない。

(2)に関しては、セションキー Kmu,ts で暗号化されているため、意味のある情報を取り出すことも出来なければ、意味のある改竄も不可能である。

(c)MU・MS間の攻撃

MU・MS間で流れる情報は、(1)MUからMSへ送られる IDmu とタイムスタンプをセションキー Kmu,ms で暗号化したもの、チケット Ticket-B をMSの公開キー pKms で暗号化したもの、MUの電子署名 sKmu を施し、MHの公開キー pKmh で暗号化したパスポート、および、(2)MSで確認が取れた後送られてくるログイン手続きメッセージである。

(1)に関しては、それぞれセションキー Kmu,ms、MSの公開キー pKms などで暗号化されているので、意味のある情報を取り出すことは出来ない。また、チケット Ticket-B の有効期限を短く設定しておくことにより、これらを利用したリプレイアタックは不可能となる。

(2)に関しては、MHでユーザ認証が確定された後のことであり、ユーザ認証プロトコルの範囲外の事柄であり、ここでは、セキュリティ上の考察は省略する。一般的には、前述同様、セションキーなどによる暗号化を施し、盗聴、リプレイアタック、なりすましなどへの対応処置を講じておかなければならない。

(d)MS・MH間の攻撃

MS・MH間を流れる情報は、(1)MUが電子署名 sKmu を施し、MHの公開キー pKmh で暗号化したパスポートと、および、(2)MHでの認証情報を、MSの公開キー pKms で暗号化したものである。

ここでは、MHに対するMUのユーザ認証の方法について述べているが、MHに対するMSのユーザ認証は、別途事前に確認されているものとする。

(1)に関しては、MUの電子署名 sKmu、MHの公開キー pKmh で暗号化されているので、意味のある情報を取り出すことも出来なければ、意味のある改竄も出来ない。また、これらを利用してリプレイアタックを試みたとしても、これだけの単独情報では、意味のある効果は得られない。

(2)に関しても、MSの公開キー pKms で暗号化されているので、意味のある情報を取り出すことも出来なければ、意味のある改竄も出来ない。また、リプレイアタックも無意味である。

4．5 認証モデルBの経路上での攻撃

(a)MU・MS間の攻撃

MU・MS間を流れる情報は、(1)MUのユーザIDの IDmu と、MUの電子署名 sKmu、MHの公開キー pKmh で暗号されたパスポート、および、(2)MHで認証された後のログイン手続きメッセージである。

(1)に関しては、MUのユーザ IDmu がそのまま流れるので、それを盗聴することにより、どのユーザがどのドメインにいるかが容易に判明される。また、これを意味のある別のユーザ ID に書き換えられる可能性がある。しかし、このIDはMHを特定するための役割もあるので、このIDが改竄されると別のMHに対して処理されることになる。ここでは、MHでの認証の後、MHから返されるログイン・プロプント・メッセージに従って、MU・MH間で再度パスワードなどによる確認が取られるので、別の不正MUが別のMHに接続されることを防止することは可能である。

パスポートは、MUの電子署名 sKmu、MHの公開キー pKmh で暗号化されているので、意味のある情報の取り出し、および、意味のある情報の改竄は不可能である。これらの情

報を盗聴した後、リプレイアタックによる「なりすまし」に対しては、パスポートの中に、タイムスタンプなどを含ませて、パスポートの有効期限を設定することにより、防止することが可能である。

(2)に関しては、MHでユーザ認証が確定された後のことであり、ユーザ認証プロトコルの範囲外の事柄であり、ここでは、セキュリティ上の考察は省略する。一般的には、前述同様、セションキーなどによる暗号化を施し、盗聴、リプレイアタック、なりすましなどへの対応処置を講じておかなければならない。

(b)MS・AS間の攻撃

MS・AS間を流れる情報は、(1)MSから送られるMSのユーザIDの IDms とTSのユーザIDの IDts、および、(2)ASから返されるMSの公開キー pKms で暗号化したセションキー Kms,ts と、TSの公開キー pKts で暗号化したチケット Ticket-A である。

(1)IDms と IDts については、公開情報なので、盗聴されても問題にはならない。2つのID をリプレイしたとしても、MU に「なりすます」ことはできない。なぜならば、AS から送られてくる情報は MU の公開鍵で暗号化されているので、それを解読できない限り、それ以降の処理が続けられないからである。

IDms を改竄してMSを「すり替える」ことは、MHとの交信に、MUから送られてきたパスポートを送らなければならないために、単純には不可能である。また、IDts を改竄して、別のTSにすり替えたとしても、チケット Ticket-A が、すり替えられたTSの公開キーで暗号化されるだけであって、この後、MSが所定のTSと交信することから、何も有効な効果は期待出来ない。

(2)については、ASから送られるものは、MSの公開キー pKms で暗号化されているので、意味のある情報は取り出すことも出来なければ、意味のある改竄も不可能である。また、ASから送り返されるものには、タイムスタンプも含まれているので、リプレイアタックも不可

能である。

(c)MS・TS間の攻撃

MS・TS間を流れる情報は、(1)MSのユーザIDの IDms と現在時刻 CurrentTime をセションキー Kms,ts で暗号化したもの、チケット Ticket-A をTSの公開キーで暗号化したものと、MHのユーザIDの IDmh、および、(2)セションキー Kms,mh とMHの公開キー pKmh で暗号化したものを、更にセションキー Kms,ts で暗号化したものである。

(1)に関しては、セションキー Kms,ts、および、TSの公開キー pKts で暗号化されたものから、意味のある情報を取り出すことも出来なければ、意味のある改竄も不可能である。

IDmh が盗聴されると、当該MHに接続しようとしているMUの存在が明らかになる危険性がある(匿名性の問題)。また、IDmh を改竄して、別のMHに「すり替える」ことは、最終的にパスワードを知っている必要があるため、単独では不可能である。

これらのいずれを使って、リプレイアタックを試みても、この後TSから送られてくる情報がセションキー Kms,ts で暗号化されているので、MSに「なりすます」ことは不可能である。

(2)に関しては、セションキー Kms,ts で暗号化されているため、意味のある情報を取り出したり、意味のある改竄は不可能である。また、これらを盗聴して、リプレイアタックを試みることは、単独では意味をなさない。

(d)MS・MH間の攻撃

MS・MH間を流れる情報は、(1)MSからMHへ送られ情報で、MSのユーザIDの IDms と現在時刻をセションキー Kms,mh で暗号化したもの、チケット Ticket-B をMHの公開キー pKmh で暗号化したものと、MUが電子署名 sKmu を施したパスポートを更に、MHの公開キー pKmh で暗号化したもの、および、(2)MHで認証されたことを示す認証確認メッセージをセションキー Kms,mh で暗号化したものである。

(1)に関しては、セションキー Kms,mh で暗号化された IDms、現在時刻から、意味のある情報を取り出すことは不可能であり、また、意味のある改竄を施すことも不可能である。また、MHの公開キー pKmh で暗号化されたチケット Ticket-B、および、MUの電子署名付きのパスポートから、意味のある情報を取り出すことは不可能である。

これらの情報をリプレイして、MSに「なりすます」ことは可能であるが、チケット Ticket-B の有効期限を短く設定することにより、リプレイの困難にすることが可能である。

(2)に関しては、セションキー Kms,mh 暗号化されているため、意味のある情報を取り出したり、意味のある改竄は不可能である。また、これらを盗聴して、リプレイアタックを試みることは、単独では意味をなさない。

4．6 結言

モーバイルコンピューティング環境(MCE)について、認証モデルA(図4.6)と認証モデルB(図4.7)の区別を明確にし、認証モデルBについてケルベロス方式と電子パスポートを組み合わせた新たなユーザ認証プロトコルGMAPを提案した。あわせて、ユーザ認証プロトコルの記述方式に提案した。更に、各モデルでの経路上における第三者による攻撃に対する安全性を比較し考察した。

認証モデルA(図4.6)は、ケルベロス方式を一貫して利用し、順次後者(サーバ)に対して前者(ユーザ)を認証する場合にあたる。モーバイルサーバとモーバイルホストは同一位置づけとみなし、モーバイルユーザ対固定環境に集約したモデルである。一方、認証モデルB(図4.7)は、モーバイルユーザ対モーバイルホストに注目し、モーバイルホストに対するモーバイルユーザの認証には、モーバイルユーザの電子パスポートを利用する。一方、マルチホッピングに関わるモーバイルサーバについて、そこでの誠意あるメッセージの中継処理を保証するためにケルベロス方式によるユーザ認証を利用するモデルである。

ユーザ認証方式GMAPでは、第三者からの攻撃を考慮して、認証に関わる情報を少なく抑えられている。モーバイルユーザは最寄りのサーバに直接無線接続し(シングルホッピング)、その後は一般的に固定ネットワーク上の複数のサーバを経由(マルチホッピング)して、目的のホストに接続される。ケルベロス方式の考え方によれば、モーバイルユーザと最寄りのサーバの間で認証局、チケットセンタを利用した第三者認証が行われるが、ここではモーバイルユーザの電子パスポート方式を提案し、無線通信の間での情報量を少なく抑えている。固定ネットワーク上での各サーバ間では、ケルベロス方式により、サーバの認証が行われ、モーバイルユーザの電子パスポートが安全に中継されて、目的のホストに先送りされる。最終的にホストで電子パスポートによりモーバイルユーザの認証が行われる。

第5章 プロトコル・シミュレータ

5．1 緒言

ユーザ認証プロトコル GMAP の頑強性(ロバストネス)を明らかにするために、認証プロトコルのメッセージ送受信動作に対し、さまざまな攻撃を発生させ安全性を確認するセキュリティシミュレータ (SS/AG) を作成する。 SS/AGの目的は、認証プロトコルの論理的正確さを照査するのではなく、設計者が気付くことなく認証プロトコルに潜む「抜け穴」(セキュリティホール)を探し出すことである。 SS/AG では、ネットワーク上での第三者からの攻撃をランダムにあるいは系統だてて発生させる事ができる。また、これらの攻撃に対する反応をプロトコルに関係する各計算機ごと、あるいは認証プロトコル全体に対して確認できるという特長を有す。

5．2 認証プロトコルの評価方法

認証プロトコルの完全性を確かめる方法として、BAN Logic に代表される数学的な手法がいくつか提案されている[Burrows-90][Glasgow-92]。このような方法の場合、プロトコルの完全性を数学的に証明でき、計算機による自動化も可能である。

しかし、このような数学的解析を用いた場合、そのプロトコルに対して、何らかの攻撃が加えられた時、どのような挙動を示すのか、どんなことが起こるのかといった実際の動作を確認することは困難である。

一方、認証プロトコルの構築の段階においては、セキュリティに関する脅威、つまり、情報の盗聴や改竄などに関して十分な安全性が保てるかどうかの検討が行われる。このような場合、プロトコルの全体や各段階を追い、その時に起こりうる状況を想定して問題がないかチェックしていくような手法が取られる。

この方法の場合、どうしても人間の思考に頼らざるを得ず、また、検証が不完全になりやすいため、プロトコルの完全性を示すことはできない。一定のアルゴリズム等が存在するわ

けではなく、計算機で自動化することは難しい。

そこで、認証プロトコルに対し、どのような攻撃を加えた時、プロトコルがどのような挙動を示すか。また、問題が起きた場合、どの程度の影響があるのか、どの程度の攻撃にまで耐えられるのか、つまり、そのプロトコルがどの程度のロバストネスを持っているのかなどを評価する方法を提案する。以下では、これらを実現するためのシミュレータについて検討する。

このシミュレータでは、プロトコルの各段階を実行し、認証を行うシミュレーション環境を作成する。そこに第三者からの攻撃を発生させ、その動作を見る事を考える。

プロトコルの安全性には、以下の二つの要素がある。

(1)暗号の安全性

(2)安全性を仮定したもの(例えば、Kerberos のチケットサーバなど)の安全性

(1)の暗号の安全性、つまり、暗号がどの程度強固であるかという点に関しては、基本的にどの暗号技術を選択するかという問題である。本研究で扱うセキュリティシミュレータは、認証プロトコルに対して攻撃が加えられた時にどのような反応を示すかを見るのが目的であり、この暗号技術の安全性の問題については言及しない。

また、(2)に関しては、セキュリティシミュレータ上で仮定した安全性が崩れた状態を仮定して、その時に何が起こるかを見ることはできる。しかし、仮定した安全性がどのようにしたら崩れるのか、また、どの程度崩れやすいのかといった点に関しては認証プロトコル以外の部分の話であり、暗号の安全性同様、セキュリティシミュレータの扱う範囲ではないと考える。

(1)に関しては、基本的にどの暗号を使うかという実装依存の問題である。しかし、暗号が解かれることはないということを前提としているのは、暗号が解かれ、認証にとってクリティカルな情報が外部に漏れることが無いことを期待しているということである。暗号化されて送られる情報は、認証にとって最重要な情報であり、外部には絶対に漏れてはいけないこ

とを意味する。

それでは、暗号化されていない情報は外部に漏れてしまっても良いのだろうか？ 基本的には、暗号化されていない情報は外部に漏れてしまっても問題ない。しかし、それらの情報は必要性があって転送されているのであり、プロトコル外部のものがそれを見た場合でも、そこから得られる情報はゼロではない。

このことから、認証プロトコルをより強固なものにするには、ネットワークの利用が少ない、つまり、認証のプロトコルの実行でネットワーク上をやりとりされるメッセージに含まれる情報量を極力減らす必要がある。

また、(2)に関しては、ネットワークに接続された計算機や何らかのオブジェクトを信用できるように高いレベルの安全性を保つのは非常に難しいことを考えなければならない。しかも、このように安全に保たれることを期待されているものが複数あった場合、その一つでも安全性が崩れるとプロトコルの完全性は保証できないことになってしまう。つまり、認証プロトコルをより強固にするには、安全に保たなければならない計算機やオブジェクトの数をなるべく少なくすれば良い。

次に、認証プロトコルの動作の各段階は、ある計算機から別の計算機へのメッセージの送信とみることができる。

送信者となる計算機では、送られるメッセージが組み立てられ、受信者となる計算機に対してネットワークを介してメッセージが送られる。受信者はそのメッセージを受け取ると、そのメッセージに特定の操作を加え、メッセージの確認を行うという作業の連続とみなすことができる。また、多くの場合、メッセージの受信者が次の段階のメッセージ送信者となる。

このような認証プロトコルでやりとりされるメッセージの作成には、

- ID や 公開鍵などの公開情報
- 秘密鍵やセッションキーなど非公開の秘密情報

が利用される。このうち後者の情報に関しては、送信者となる計算機全てが利用できる情

報とは限らず、プロトコル中のある段階での送信者の動作をシミュレートする場合、任意のメッセージを自由に作成して良いわけではない。

メッセージに利用される秘密情報をその送信者が知っているか、言いかえれば、送信者がそれまでのプロトコルの(シミュレーションの)動作の結果や自分がもともと保持していた情報などから、その送信メッセージに含まれる秘密情報、また、暗号化などに使用される秘密鍵などを所有しているかを確認することが必要になる。

一方、受信者のメッセージへの操作は大きく以下のように分けられる。

(1) 復号化(電子署名の確認)

(2) タイムスタンプの有効期限の確認

(3) nonce の確認

このうち (1) に関しては、送信者の場合と同様、復号化(電子署名の確認)に使われる鍵を受信者が持っているかどうかが問題になる。また、受信者はメッセージに含まれるもののうち以後の認証のプロセスに必要になる情報を記録しておく必要がある。

5.3 ユーザ認証プロトコルに対する脅威

ユーザ認証プロトコルに対し、外部の第三者が悪意を持って行う攻撃を分類すると、

・情報の盗聴

ネットワーク上でやりとりされるメッセージをそのまま取り込み、盗み見る行為

・情報の再送

盗聴などで得た正常なメッセージを別のタイミングで再び送信する行為

・情報の改竄

ネットワーク上でやりとりされる情報の一部を受信者に届く前に変更してしまう行為

・情報の横取り

メッセージをネットワーク上で盗み取って、相手に届かなくする行為

・偽メッセージの送信

不正なメッセージを新たに作成し、受信者に送りつける行為

が考えられる。これらの攻撃は、送信者から受信者に対してメッセージが送られる時、つまり、メッセージがネットワーク上を転送されている時に加えられる。

実際に、目的をもって攻撃を仕掛けてくるユーザはこれら単純な攻撃を組合せ、プロトコルの問題点を突き不正を行おうとする。また、完全性が証明されているプロトコルの場合でも、なんらかの理由によりその完全性の前提としている条件の一部が崩れて(崩されて)しまった場合、例えば、暗号化に用いる鍵の選択の不備などにより、容易に暗号が解読できてしまうような場合などにも、不正が行えてしまう可能性がある。

これら以外にプロトコルに関係する計算機自身がなにか悪意を持って不正なメッセージを流したり、不当なデータ操作を行うといったような、内部からの攻撃も考えられる。

分散環境で第三者認証を行う場合、最小限のある一定の計算機を「信用できる」計算機として定義する。「信用できる」計算機はあらゆる面で安全性が保たれており、無条件に信用できるものと考える。

内部からの攻撃を考えた場合、状況は 2 通り考えられる。一つは、プロトコルに関係する一般の計算機から不正が行われようとした場合、もう一つは、この「信用できる」計算機が不正を行おうとした場合である。このうち、「信用できる」とした計算機の動作は無条件に信用されるので、ここからの攻撃が起こった場合は防ぐ事ができない。

5. 4 プロトコル・シミュレータ SS/AG

セキュリティシミュレータ SS/AG(Security Simulator with Attack Generator) は主に二つの要素

・認証プロトコルに関係する各計算機の動作をシミュレートするホストシミュレータ部

・プロトコルに対するさまざまな攻撃を発生させるアタックジェネレータ

から成る。

図.5.1の中央に見えるホストシミュレータは、どの計算機の動作を行うかやメッセージの形式、メッセージに加える操作を入力として受け取り、プロトコルの各段階の計算機の動作を行う。

ホストシミュレータで作成されたメッセージは通信経路にあたるメッセージバッファに移される。 アタックジェネレータは、メッセージバッファに置かれているメッセージに対して操作を行うことにより、認証プロトコルに対する様々な攻撃を発生させる。 これらの動作をシミュレートする各計算機ごと、また、プロトコル全体の動作に対して記録し、どのような攻撃が起きた場合にプロトコルがどのような反応を示すのか、またプロトコルに問題は無いかなどを見る。

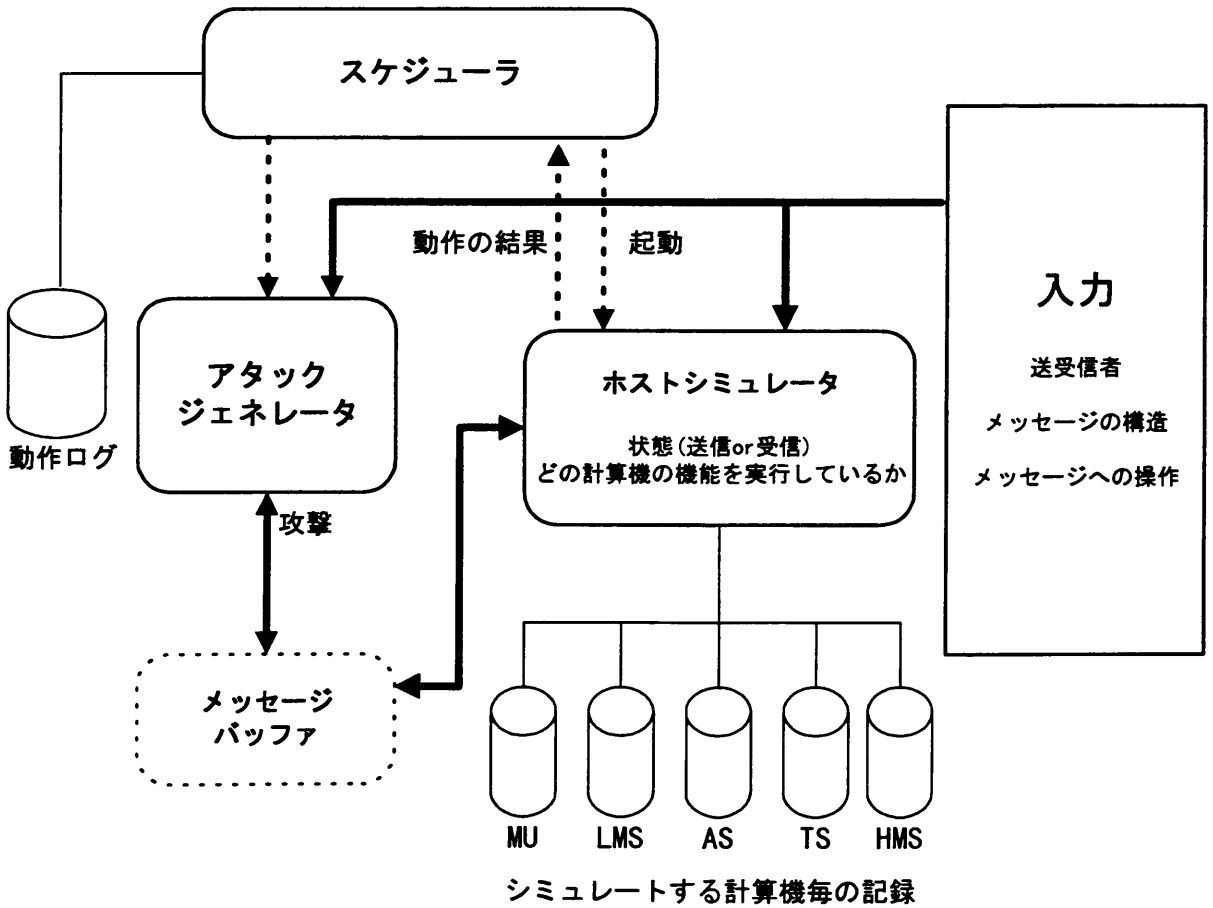

図5.1 セキュリティシミュレータ SS/AG

5．4．1 ホストシミュレータ

前述のように、ユーザ認証のプロトコルは、

・送信状態

指定された形式でメッセージを組み立てる。

・受信状態

メッセージを受け取り、そのメッセージに復号化などの操作を加える。

という動作の連続となる。

この送信者、受信者は常に固定されているわけではなく、ある計算機がプロトコルの途中で送信者になったり、受信者になったりする。 そこで、SS/AG ではホストシミュレータ部がプロトコルの各段階に従って各計算機の動作、つまり送信時のメッセージの組み立て、および受信時のメッセージに対する操作をシミュレートする。

ホストシミュレータ部はプロトコルに関係する計算機ごとに、それまでの動作から得られ、その計算機が知っている情報を保存する必要がある。 ホストシミュレータ部の動作は (1)送信状態か受信状態か、(2)どの計算機の動作か、(3)メッセージの構成、内容、そのメッセージに対する操作、によって決まる。

送信状態では、ホストシミュレータ部は送信メッセージの作成を行う。 メッセージに含まれる情報には、一般に公開されている情報(ID や公開鍵など)と公開されていない秘密情報がある。このうち、秘密情報は当然それを知っているものしか利用できない。

また、SS/AG では、メッセージの一部に対して暗号化の指示があった場合でも、実際の暗号化は行わず、その部分が暗号化された部分である事を示すだけになっている。これは、SS/AG が第三者によって実際の暗号の解読が可能かどうかといったような事実の評価を目的としたものではないためで、実際の暗号化、復号化といった作業は不要だからである。

送信状態で作成されたメッセージは、ネットワークの経路にあたるメッセージバッファに

移される。 受信状態では、メッセージバッファから取り出したメッセージに指示された操作を行なうメッセージに対して加えられる操作は以下である。

・復号化

メッセージのうち、指定された鍵で暗号化された部分の復号化

・比較

メッセージの指定された部分と特定の値を比較する。

5．4．2 アタックジェネレータ

前述のように、認証プロトコルに対する攻撃には、内部からの攻撃と外部からの攻撃が考えられる。さらに、内部からの攻撃は「信用できる」計算機からのものとそれ以外のものに分ける事ができる。このうち、「信用できる」計算機からの攻撃に関しては、認証プロトコルの実行によって検出する事は不可能である。そのため SS/AG においては、この種の攻撃は扱わない。

アタックジェネレータは認証プロトコルに対する外部からの攻撃を発生させることを行う。ホストシミュレータ部の送信状態で作成されたメッセージは一旦メッセージバッファに格納される。アタックジェネレータはこのメッセージバッファに格納されたメッセージに対して、操作を行う。

アタックジェネレータは認証プロトコルへの攻撃を実現するために、以下の基本機能を持つ。

・ 盗聴

メッセージバッファからアタックジェネレータ内部にあるメッセージ

を納めておくバッファにコピーする。

・ 送信

アタックジェネレータの内部バッファからメッセージバッファへ指定

されたメッセージをコピーする。

・ 変更

アタックジェネレータの内部バッファにあるメッセージの内容を変更する。

認証プロトコルに対する外部からの攻撃は、以下のように基本機能の組合せで実現できる。これを基本攻撃と呼ぶ。

情報の盗聴 = ``盗聴''

情報の再送 = ``盗聴'' と ``送信''

情報の改竄 = ``盗聴'' + ``変更'' + ``送信''

情報の横取り = ``盗聴'' + (空情報の)``送信''

偽メッセージの送信 = ``変更'' + ``送信''

実際の攻撃者は、さらに、これらを様々な形に組み合わせて攻撃を行う。アタックジェネレータで、こういった様々な攻撃を行おうとした場合、アタックジェネレータによるメッセージの``変更''が問題になる。

攻撃者がメッセージに改竄を加えようとする場合、(a)メッセージを単純なビットの連続としてとらえ、メッセージの構造を知らずに改竄を行う、(b)メッセージの構造を知っていて構造に沿って改竄を行う、という二通りの場合がある。

攻撃者が何かの目的をもって攻撃を行う場合、つまり何らかの意味のある攻撃を加えようとした場合はメッセージの構造を知り、後者のパターンを行う。

SS/AG では、この目的を持った攻撃を実現するため、アタックジェネレータで発生させるメッセージの ``変更'' は後者の方法を実現させる。従って、アタックジェネレータは攻撃を発生させる時点でのメッセージの構造を入力として与えられる。

次に、攻撃発生の制御方法について述べる。 攻撃者が何らかの目的を持って攻撃を発生させる場合、前述のプリミティブな攻撃を様々に組合せ、的確なタイミングで発生させ

ることを行う。攻撃者はプロトコル全体を把握し、どのタイミングでどのようなことを行えば良いかを考えるわけである。

アタックジェネレータで攻撃を発生させる場合、このどのタイミングでどの攻撃を発生させるのか、攻撃の発生をどのように制御するのかが問題になる。 アタックジェネレータにおいて、攻撃者が行う全ての攻撃を完全にシミュレートするのは不可能である。SS/AG では、(1)基本攻撃のランダムな組み合わせ、および、(2) プロトコルの設計者が想定した攻撃に対して、認証プロトコルがどのような反応を示すか、意図した通りの動作を行うかを検証できる。 (2) については、あらかじめスクリプトに記述しておき、それをアタックジェネレータに入力することによって、攻撃発生を制御する。

5. 4. 3 SS/AG の実行例

SS/AG に次のような認証プロトコルを与えた場合の動作の例を示す。

(1) A → S : A, B, N_A

(2) S → A : $\{N_A, B, K_{AB}, (K_{AB}, A)_{K_B}\}_{K_A}$

(3) A → B : $\{K_{AB}, A\}_{K_B}$

(4) B → A : $\{N_B\}_{K_{AB}}$

(5) A → B : $\{N_B - 1\}_{K_{AB}}$

ここで、A はホストの ID、K_X は X の鍵、N_X はランダムな数を表している。例えば、上記(3) は A から B に鍵 K_{AB} と ID A を 鍵 K_B で暗号化したものを送ることを表す。

このプロトコルをシミュレータに与え、(3)のメッセージを再送するスクリプトをアタックジェネレータに入力し、これに従って攻撃を生成したとすると以下の出力例が得られる。

Attacker to B : (Replay) S_b (O_ab, I_A)

```
  Success  on  B
B  to  A      :        O_ab  (N_b)
  Success  on  A
A  to  B      :        O_ab  (N_b  -  1)
  Success  on  B
Auth Complete  :  (Replay)
```

図5.2. シミュレータの出力例

この出力において、S_b はBの秘密鍵、O_ab は、A と B の間で一時的に利用されるセッションキー、I_A は、AのID、N_b はBによって生成されたランダムな値を表わしている。この出力の最初の行は、アタッカからBに、メッセージ "S_b (O_ab, I_A)" の不正再送が行われたこと表わしている。2行目は、メッセージ "S_b (O_ab, I_A)" を受け取り、Bがメッセージに与えられた操作を加えて、メッセージの正当性を判断した結果である。メッセージ " S_b (O_ab, I_A)" は、アタッカから送られたものであるが、正当なメッセージを受け取ったと判断されている。以下、プロトコルに従って認証動作が進んでいき、最後の行が、プロトコル全体を通して、認証が成功したこと、更に、再送攻撃があったことを示している。

上記プロトコルの場合、(3) で送られたメッセージが攻撃モジュールによって再送されても認証が成功してしまうという結果が得られることになる。このことから、上記プロトコルでは(3)で送られるメッセージにランダムな数を入れたり、タイムスタンプを入れたりすることによってプロトコルを補強する必要がある事が判断できる。

また、ランダムな300回の攻撃に対して、51回について攻撃が成功したことが判定された。

次に、GMAP に対して、攻撃を発生させた例を示す。

Attacker to LMS : (Repley) P_LMS(O_LMS_TS),P_TS(I_LMS,I_TS,O_LMS_TS,T_1)

Success on LMS : P_LMS(O_LMS_TS),P_TS(I_LMS,I_TS,O_LMS_TS,T_1)

LMS to TS : O_LMS_TS(I_LMS,T_2), P_TS(I_LMS,I_TS,O_LMS_TS,T_1), I_HMS

check fail : T_1

Fail on TS : O_LMS_TS(I_LMS,T_2), P_TS(I_LMS,I_TS,O_LMS_TS,T_1), I_HMS

Auth Fail : (Replay)

この例では、attacker からLMSに対してメッセージの再送が行われている。その結果、最初の2行を見るとLMSでは正常なメッセージとして処理されてしまっているが、TSが受け取った時点で、メッセージ中に含まれる T_1 というタイムスタンプをチェックした結果、メッセージが不当な物であると判断され認証に失敗している。 つまり、GMAPにおいては、ASからLMSに送られる、上記のメッセージだけが再送された場合、認証プロトコル中で、それを不当なメッセージであると判断できることがわかる。

LMS to TS : O_LMS_TS(I_LMS,T_2), P_TS(I_LMS,I_TS,O_LMS_TS,T_1), I_HMS

Attacker to TS : (Modify) O_LMS_TS(I_LMS,T_2),P_TS(I_LMS,I_TS,O_LMS_TS,T_1),I_HMS2

Success on TS : O_LMS_TS(I_LMS,T_2),P_TS(I_LMS,I_TS,O_LMS_TS,T_1),I_HMS2

TS to LMS : O_LMS_TS(O_LMS_HMS2,P_HMS2(I_LMS,I_HMS2,O_LMS_HMS2))

Success on LMS : O_LMS_TS(O_LMS_HMS2,P_HMS2(I_LMS,I_HMS2,O_LMS_HMS2))

LMS to HMS : O_LMS_HMS2(I_LMS,T_3),P_HMS2(I_LMS,I_HMS2,O_LMS_HMS2),P_HMS(S_MU(M_PASS))

decode fail : P_HMS2

Fail on HMS : O_LMS_HMS2(I_LMS,T_3),P_HMS2(I_LMS,I_HMS2,O_LMS_HMS2),P_HMS(S_MU(M_PASS))

Auth Fail : (Modify)

これは、アタッカがLMSからTSへ送られたメッセージの一部を改変した例である。元々の メッセージはTSに対してLMSとHMS間の通信に使うチケットとセッション鍵を要求しているが、HMSのIDを意図的にHMS2に書き換える事により、LMSとHMS2間の通信を行うための要求があったというメッセージになっている。

このメッセージは、TSにとっては、正常な要求なので、正常に処理される(3行目)。しかし、受け取ったメッセージは LMSとHMS2間でやりとりを行うためのチケットを含み、HMS2のみが入手できるようにHMS2の公開鍵で暗号化されている(4 行目)。

LMSにとっては、TSから受け取ったメッセージは、HMSへメッセージを送るための要求であり、受け取ったメッセージを処理すると、メッセージを作成して、HMSに向けて送信することになる(6 行目)。

HMSでは受け取ったメッセージから、セッション鍵を取り出すため、メッセージの復号化を試みる。しかし、セッション鍵を含むメッセージは、HMS2の公開鍵で暗号化されているため、復号することができない(7 行目)。従って、HMSはセッション鍵を取り出す事ができず、正常なメッセージとして処理できないことから認証は失敗している。つまり、TSへの要求を改竄して、LMSとHMS2の間という別の計算機間の要求であるように書き換えただけでは、不正を行う事はできないということである。

また、ランダムな300回の攻撃に対して、いずれも攻撃が失敗したことが判定された。

5.5 結言

ユーザ認証プロトコルGMAPを評価する方法について述べ、ユーザ認証プロトコルGMAPのロバストネスに関する検討を行い、ネットワーク上を流れる認証に関する情報量を

極力小さくし、安全に保たなければならない計算機の数をなるべく少なくすることが、認証プロトコルのロバストネスを高めることを考察した。

さらに、ユーザ認証プロトコルを評価するためのシミュレータに関して検討し、外部からの第三者攻撃に対するユーザ認証プロトコルの安全性を確認するために、プロトコルシュミレータSS／AGを作成し、ユーザ認証プロトコルの耐攻撃性(ロバストネス)の評価を概括した。まず、プロトコルを構成する基本メッセージである当事者同士で交信されるメッセージに着目し、メッセージの暗号化による静的な安全性の確認を行い、各メッセージの組み合わせによるプロトコルの動的な安全性について考察した。SS／AGは、プロトコルを構成するメッセージを1文ずつプロトコルバッファから取り出して解析するプロトコルアナライザ部と、第三者による攻撃をシミュレートするプロトコルアタッカ部から構成される。 ネットワーク上での第三者からの攻撃をランダムにあるいは系統だてて発生させる事により、認証プロトコルに潜む「抜け穴」(セキュリティホール)を探し出すことができる。

第6章　結論

ネットワーク、携帯型端末の技術進歩の結果、従来の計算機が固定設置された固定計算環境(FCE)に加えて、計算機を携行して人の移動と共に計算環境が移動する移動計算環境(MCE)が可能になってきた。FCEでは、計算機の設置と共にネットワーク自体も固定されるが、MCEでは、無線を利用して何時でも誰でも誰とでも、利用者と利用する場所を特定しない環境になっている。従ってMCEでは新たな課題として、正当な利用者であることをネットワークに入る前に確認しなければならない。本論文は、この問題、すなわち、MCEにおけるユーザ認証に関して、MCEモデル、ユーザ認証プロトコルGMAP、プロトコル検証シミュレータSS／AGを提案し考察した。

第1章では本研究の背景、目的を述べた。第2章では本研究に関連する従来の研究動向を述べた。最近のパソコンの小型軽量化、および、ネットワーク技術、とりわけ携帯電話、無線技術の進歩の結果として、MCE環境の考え方が一般的になってきた。従来からのFCE環境で研究されてきた課題は、そのままMCE環境に敷延することの可否については、まだまだ議論の残るところである。ユーザ認証については、従来からFCE環境を基本にしたUNIXベースのネットワークでの研究が活発に行われてきたが、常時ネットワーク接続された計算機を前提にしたFCEに代わって、必要に応じて都度ネットワークに接続するMCE環境においては、ネットワークに入る時点において、正当なユーザであることが確認されなければならない。

第3章ではMCE環境におけるユーザ認証の観点から、従来のFCE環境を拡張して、MCE環境のモデルを提案した。FCE環境の場合、利用者はネットワーク接続され、かつ、常時ホストに捕捉された端末に赴き、ログイン操作によりホスト計算機にアクセスする。一方、モーバイルユーザが携行する移動端末は、通常はネットワークに接続されておらず、必要に応じて逐次移動端末に備えられた無線機能を利用してネットワークに接続する。更に直接目的のホストに接続するのではなく、まず最寄りのサーバに無線接続し、自分が登

録された目的のホストに接続を託する。この場合ユーザは登録先の目的ホストを指定することなく、単にユーザ識別子をサーバに伝えるだけで、サーバがユーザ識別子から簡単に接続先のホストを割り出せるような、ユーザ識別子方式を提案した。

第4章では、MCE環境モデルの上で、ケルベロス方式と電子パスポートを組み合わせた新たなユーザ認証プロトコルGMAPを提案した。このユーザ認証方式では、第三者からの攻撃を考慮して、認証に関わる情報を少なく抑えられている。本研究のモデルでは、モーバイルユーザは最寄りのサーバに直接無線接続し(シングルホッピング)、その後は一般的に固定ネットワーク上の複数のサーバを経由(マルチホッピング)して、目的のホストに接続される。ケルベロス方式の考え方によれば、モーバイルユーザと最寄りのサーバの間で認証局、チケットセンタを利用した第三者認証が行われるが、ここではモーバイルユーザの電子パスポート方式を提案し、無線通信の間での情報量を少なく抑えている。固定ネットワーク上での各サーバ間では、ケルベロス方式により、サーバの認証が行われ、モーバイルユーザの電子パスポートが安全に中継されて、目的のホストに先送りされる。最終的にホストで電子パスポートによりモーバイルユーザの認証が行われる。メッセージの暗号化には公開鍵方式を使用し、モーバイルユーザの電子パスポートは、モーバイルユーザの秘密鍵で暗号化したものに、更にホスト公開鍵で暗号化を施すが、モーバイルユーザとホストとの特別な関係を考慮すれば、公開鍵方式による二重暗号化に伴う解読不能問題を回避できるように、予め両者の公開鍵と秘密鍵を設定することが可能である。

第5章では、外部からの第三者攻撃に対するユーザ認証プロトコルの安全性を確認するために、プロトコルシュミレータSS／AGを作成し、ユーザ認証プロトコルの耐攻撃性(ロバストネス)の評価を概括した。まず、プロトコルを構成する基本メッセージである、当事者同士で交信されるメッセージに着目し、メッセージの暗号化による静的な安全性の確認を行い、各メッセージの組み合わせによるプロトコルの動的な安全性について考察した。SS／AGは、プロトコルを構成するメッセージを1文ずつプロトコルバッファから取り出して解

析するプロトコルアナライザ部と、第三者による攻撃をシミュレートするプロトコルアタッカ部から構成される。

今後は、本研究によるユーザ認証プロトコルGMAP、および、プロトコルシミュレータSS/AGの改良を重ね、プロトコルの耐攻撃性(ロバストネス)を定量的に把握するための指標を導出し、より安全なプロトコルを設計するための環境に供する。

ユーザ認証プロトコルGMAPについては、暗号化を施す部分の最適設定、および、第三者攻撃を前提にしたユーザ認証プロトコルの動的な構成法などを考察していく。プロトコルシミュレータSS／AGについては、特定当事者間のユーザ認証プロトコルの評価に加えて、複数当事者間で、かつ、それぞれ異なるユーザ認証プロトコルによるユーザ認証が同時・不規則に動作している環境における第三者攻撃を想定した環境にも適用できるようにしていく。こうした拡張に合わせて、ユーザ認証プロトコルに求められる特性として、第三者により容易に攻撃されにくいユーザ認証プロトコルの構成法、また、第三者攻撃に耐えうるためにユーザ認証プロトコルに求められる必要条件を定量的に評価する指標の導出を試みる。合わせて、ユーザ認証プロトコルと一緒に考慮されるべきサーバ認証プロトコルについても、同様な観点から考察を試みたい。

更には、モーバイルコンピューティングで繰り広げられるビジネス環境、たとえば、インターネットなど、ネットワーク下での電子商取引(エレクトリックコマース)などへの適応性についても、実験を試み、評価していきたい。インターネット環境下におけるエンド・エンドに加えて、インターネット環境に閉域構成されたエクストラネットとの組み合わせ環境を想定した、認証プロトコル、ひいては、ファイヤウォールなどとの比較において、第三者攻撃に対するセキュリティの観点から比較を試みたい。

謝辞

本研究において、直接種々懇切丁寧なご指導とご鞭撻を頂いた静岡大学情報学部助教授渡辺尚博士に深く感謝申し上げる。

本論文を纏める過程で、種々適切なご指導とご鞭撻を頂いた静岡大学教授浅井秀樹博士、静岡大学教授市川朗博士、静岡大学教授福田明博士、静岡大学教授曽我正和博士、静岡大学教授鈴木淳之博士、静岡大学教授阿部圭一博士、および、静岡大学教授水野忠則博士に深く感謝申し上げる。

本研究の契機を直接啓発して頂いた三菱電機株式会社常務取締役武藤達也氏、静岡大学大学院博士課程への社会人入学の機会を与えて頂いた株式会社インテック代表取締役社長中尾哲雄氏、日頃励まして頂いた株式会社インテック代表取締役副社長河野隆一氏に深く感謝申し上げる。

本研究に直接協力頂いた静岡大学情報学部助教授佐藤文明氏、助手西田正勝氏をはじめとする諸先生、飯田登氏、石川睦氏をはじめとする渡辺研究室および水野研究室の学生の皆様に深く感謝申し上げる。

最後に、在宅の研究作業に協力を強いた家族に感謝する。

付録1. フェルマーの小定理

フェルマーの小定理の証明方法の一つとして、二項係数による方法について述べる。

p を素数、a を整数とするとき、 $p \nmid a$ (整数aが、素数pで割り切れない)ならば、$a^{p-1} \equiv 1 \pmod{p}$ である。

まず、二項係数 ${}_pC_i$ について、

$$ {}_pC_i = \frac{p!}{i! \times (p-i)!} $$

$$ = \frac{p(p-1) \cdots (p-i+1)}{1 \cdot 2 \cdots i} \qquad (1 \le i \le p) $$

から、pが素数で、iが、$i < p$ を満たす整数のとき、分母は、pで割り切れないので、$1 \le i < p$の範囲で、${}_pC_i$は、pで割り切れることになる。ただし、${}_pC_0 = 1$、${}_pC_1 = 1$ とする。

また、$a = 2$ の場合、

$$ 2^p = (1+1)^p = {}_pC_0 + {}_pC_1 + \cdots + {}_pC_{p-1} + {}_pC_p $$
$$ \equiv (1+1) \pmod{p} $$
$$ \equiv 2 \pmod{p} $$

となる。一般的に、$(a-1)^p \equiv a-1 \pmod{p}$ と仮定すれば、

$$ a^p = \{(a-1)+1\}^p $$
$$ = (a-1)^p + {}_pC_1(a-1)^{p-1} + {}_pC_2(a-1)^{p-2} + \cdots\cdots + {}_pC_{p-1}(a-1) + 1 $$
$$ \equiv (a-1) + 1 \pmod{p} $$
$$ \equiv a \pmod{p} $$

が導かれ、$a(a^{p-1}-1)$は、素数pで割り切れることになる。

一方、整数aは、素数pで割り切れないので、結局、$a^{p-1}-1$は、素数pで割り切れることになり、フェルマーの小定理: $a^{p-1} \equiv 1 \pmod{p}$ が得られる。

付録2．第3章の結果によるフェルマーの小定理の証明

第3章のユーザID導出で得られた 3.5 式を用いて、フェルマーの小定理を証明する。

$$2^n = \sum_{n=(i,j)} K_i = \sum_{n=(i,j)} i \times k_i \qquad (3.5)$$

上の式は、2を基数とした場合であるが、これを一般化して、m の場合ついて書き直すと次のようになる。

$$m^n = \sum_{n=(i,j)} K_{m_i} = \sum_{n=(i,j)} i \times k_{m_i} \qquad (3.5')$$

たとえば、$m=10$、$n=2$ の場合、00から99までの100個(＝10^2)の10進数について、0 から 9 の1桁エントリを2個繋げた数は、00、11、22、33、44、55、66、77、88、99 の10個である。また、01 と 10、02 と 20、…、12 と 21、13 と 31、…、79 と 97、89 と 98 のように、巡回すると同じ数の組は、45個ある。

$$10^2 = 1 \times k_{10_1} + 2 \times k_{10_2} = 1 \times 10 + 2 \times 45$$

n が素数の場合、因数は、1と n であることから、

$$m^n = 1 \times k_{m_1} + n \times k_{m_n}$$

k_{m_1} は、0から m - 1までの m 個の数を、それぞれ n 個繋げた数に対応することから、$k_{m_1} = m$ である。従って、

$$m^n = 1 \times m + n \times k_{m_n}$$

$$m\ (\ m^{n-1}\ -\ 1\ )\ =\ n\ \times\ k_{m_n}$$

これより、nが素数で、nとmが互いに素であるならば、m^{n-1} - 1 は、nで割り切れることになり、フェルマーの小定理である、$m^{n-1} \equiv 1 \pmod{n}$が導き出される。

参考文献

[Abadi-96] Martin Abadi and Roger Needham,"Prudent Engineering Practice for cryptographic Protocols",IEEE Trans.on Software Enfineering, Vol.22,No.1, pp.6-15, Jan.1996.

[ACM-95] Mobile Computing and Networking, ACM,1995.

[Aura-97] Thomas Aura, "Strategies against Replay Attacks", 10th IEEE Computer Security Foundations Workshop, pp.59-68, Jun.1997.

[Bird-92] R.Bird,I.Gopal,A.Herzberg,P.Janson,S.Kutten,R.Molva and M.Yung, "Systematic design of a family of attack-resistant authentication protocols", IEEE JSAC Special Issue on Secure Communication, 1992.

[Black-96] Uyless Black ,"Mobile and Wireless Networks", Prentice Hall PTR, 1996.

[Boyd-96] Colin Boyd,"A Class of Flexible and Efficient Key Managemnt Protocol", 9th IEEE Computer Security Foundations Workshop, pp.2-8,June 10-12,1996.

[Brodsky-97] Ira Brodsky ,"Wireless Computing - A Manager's Guide to Wireless Networking", Van Nostrand Reinhold, 1997.

[Brown-95] D.Brown, "Techniques for Privacy and Authentication in Personal Communication System", IEEE Personal Communications, Vol.2,No.4, pp.6-10,

Aug.1995.

[Burrows-89] M.Burrows,M.Abadi,and R.M.Needham,"A logic of authentication", Proc. Royal Soc.London A,vol.426,pp.233-271,1989.A preliminary version appeared as Digital Equipment Corporation Systems Research Center,report no.39,Feb.1989.

[Burrows-90] M.Burrows,M.Abadi and R.Needham,"A Logic of Authentication." ACM Trans.Computing Systems,Vol.8,pp.18-36,February 1990.

[Catherine-96] Catherine Meadows,"Language Generation and Verification in the NRL Protocol Analyzer",9th IEEE Computer Security Foundations Workshop, pp.48-61,June 10-12,1996.

[CCITT-88] CCITT Blue Book,Recommendation X.509 and ISO 9594-8:The Directry-Authentication Framework.Geneva,Mar.1988.

[Clifford-95] B.Clifford Neuman and Glen Zorn,"Integrating One-time Password with Kerberos", Internet Draft ietf-cat-kerberos-passwords-01,Apr.1995.

[Cuppens-96] Frederic Cuppens and Claire Saurel,"Specifying Security Policy: A Case Study",9th IEEE Computer Security Foundations Workshop,pp.123-134,June 10-12, 1996.

[D'Angelo-94] Diana M.D'Angelo, Bruce McNair and Joseph E.Wilkes, "Security in

Electronic Messaging Systems", AT&T Technical Journal, pp.7-13, May/June 1994.

[Dayem-97] Rifaat A.Dayem ,"Mobile Data & Wireless LAN Technologies", Prentice Hall PTR, 1997.

[Denning-81] D.E.Denning and G.M.Sacco,"Timestamps in key distribution protocols", CACM, vol.24, no.8, pp.533-536, Aug.1981.

[Focardi-95] R.Focardi,R.Gorrieri and V.Panini,"The Security Checker : A Semantics-based Tool for the Verification of Security Properties",8th Computer Security Foundations Workshop,pp.60-69,June 13-15,1995.

[Focardi-96] Riccardo Focardi,"Comparing Two Information Flow Security Properties",9th IEEE Computer Security Foundations Workshop,pp.116-122,June 10-12,1996.

[Fox-96] Armando Fox and Steven D.Gribble, "Security on the Move : Indirect Authentication Using Kerberos", Mobile Computing and Networking(MOBICOM'96), pp.155-164, Nov.1996.

[Frankel-95] Yair Frankel et al, "Security Issues in a CDPD Wireless Network", IEEE Personal Communications, Vol.2,No.4, pp.16-27, Aug.1995.

[Gibson-96] Jerry D.Gibson, "The Mobile Communications Handbook", CRC Press,Inc.,

1996.

[Glasgow-92] J.Glasgow,G.MacEwan and P.Pananageden,"A logic for Reasoning about Security." ACM Trans.Computing Systems,Vol.10,No.3,pp.265-310,1992.

[Gollmann-96] Dieter Gollmann,"What do we mean by Entity Authentication?",1996 IEEE Symposium on Security and Privacy,pp.46-54,May 6-8,1996.

[Gong-95] Li Gong,"Optimal Authentication Protocols Resistant to Password Guessing Attacks",8th Computer Security Foundations Workshop,pp.24-29,June 13-15,1995.

[Hardy-60] G.H.Hardy and E.M.Wright, "An Introduction to the Theory of Numbers", Oxford Press, 1960.

[Heintze-96] Nevin Heintze and J.D.Tygar,"A Model for Secure Protocols and Their Compositions",IEEE Trans.on Software Enfineering,Vol.22,No.1, pp.16-30,Jan.1996.

[Hickmann-94] K.E.B.Hickman,"The SSL protocol",RFC,Netscape Communications Corp.,version of Oct.31,1994.

[Hickmann-95] K.E.B.Hickman and T.Elgamal,"The SSL Protocol",Internet Draft,Netscape Communications Corp.,version of Jun.1995.

[Hoare-85] C.A.R.Hoare, "Communicating Sequential Processes", Prentice-Hall, 1985.

[Imielinski-96] Tomasz Imielinski and Henry F.Korth, "Mobile Computing", Kluwer Academic Publishers, 1996.

[Jaeger-95] Trent Jaeger and Atul Prakash,"Implementation of a Discretionary Access Control Model for Script-based Systems",8th Computer Security Foundations Workshop,pp.70-84,June 13-15,1995.

[Kohl-93] J.Kohl, B.C.Neuman, "The Kerberos Network Authentication Service"(Version 5), Internet RFC 1510, Sep.1993.

[Kohl-94] John T.Kohl, B.Clifford Neuman and Theodore Y.T'so, "The Evolution of the Kerberos Authentication System", Distributed Open Systems,pp.78-94,IEEE Computer Society Press,1994.

[Kohnfelder-78] L.M.Kohnfelder, "On the Signature Reblocking Problem in Piblic-Key Crypto-systems", Comm.ACM, pp.179, Vol.21, No.2, Feb.1978.

[Lowe-96] Gavin Lowe, "Some New Attacks upon Security Protocols",9th IEEE Computer Security Foundations Workshop,pp.162-169,June 10-12,1996.

[Lowe-97a] Gavin Lowe, "Casper: A Compiler for the Analysis of Security Protocols", 10th IEEE Computer Security Foundations Workshop, pp.18-30, Jun.1997.

[Lowe-97b] Gavin Lowe, "A Hierarchy of Authentication Specifications", 10th IEEE Computer Security Foundations Workshop, pp.31-43, Jun.1997.

[Lu-89] W.P.Lu and M.K.Sundareshan,"Secure communication in internet environment: A hierarchical key management scheme for end-to-end encryption",IEEE Trans.on Comm.vol.37,no.10,pp.1014-1023,Oct.1989.

[MacWilliams-77] F.J.MacWilliams and N.J.A.Sloane, "The Theory of Error-Correcting Codes", North-Holland Punbling Co., 1977.

[Mao-93] W.Mao and C.Boyd,"Towards formal analysis of security protocols", Proc.Computer Security Foundations Workshop VII,pp.147-158,1993.

[Mao-95] Wenbo Mao,"An Augmentation of BAN-Like Logics",8th Computer Security Foundations Workshop,pp.44-56,June 13-15,1995.

[Miller-87] S.P.Miller,B.C.Neuman,J.I.Schiller,and J.H.Saltzer,"Kerberos authentication and authorization system",Project Athena Technical Plan,Section ".2.1,MIT,Jul.1987.

[Miller-88] S.P.Miller, B.C.Neuman, J.I.Schiller, and J.H.Saltzer, "Section E.2.1:Kerberos Authentication and Authorization System"(Version 4), Project Athena Technical Plan, MIT Project Athena, Cambridge, Massachusetts, Oct.1988.

[Milner-89] Robin Milner, "Communication and Concurrency", Prentice-Hall,1989.

[Mitchell-97] John C.Mitchell, Mark Mitchell and Ulrich Stern, "Automated Analysis of Cryptographic Protocols Using Murϕ", 1997 IEEE Symposium on Security and Privacy, pp.141-151, May 1997.

[Molva-94] Refik Molva, Didier Samfat and Gene Tsudik, "Authentication of Mobile User", IEEE Network,pp.26-34,Mar./Apr.1994.

[Muftic-94] Sead Muftic and Morris Sloman,"Security architecture for distributed systems", Comupter Communications,Vol.17,No.7,Jul.1994.

[Needham-78] R.M.Needham and M.D.Schroeder,"Using encryption for authentication in large networks of computer",CACM,vol.21,no.12,pp.993-999,Dec.1978.

[Needham-87] R.M.Needham, M.D.Schroeder, "Using Encryption for Authentication in Large Networks of Computers", Communications of the ACM, Vol.21,No.12, pp.993-999, Dec.1987.

[Needham-93] R.M.Needham,"Cryptography and secure channels",Distributed Systems, 2nd edition,S.Mullender,ed.,pp.231-241,ACM Press,1993.

[Neuman-94] B.Clifford Neuman and Theodore Ts'o,"Kerrberos:An Athentication Service for Computer Networks", IEEE Communication Magazine,Vol.32,No.9,pp.33-38,Sep.1994.

[Neuman-95a] B.C.Neuman, Brian Tung, and John Wray, "Public Key Cryptography for Initial Authentication in Kerberos", Internet Draft ietf-cat-kerberos-pk-init-00, Mar.1995.

[Neuman-95b] B.Clifford Neuman and Glen Zorn, "Integrating One-time Password with Kerberos", Internet Draft ietf-cat-kerberos-password-01, Apr.1995.

[Otway-87] D.Otway and O.Rees,"Efficient and timely mutual authentication",Operating Systems review,vol.21,no.1,pp.8-10,Jan.1987.

[Patel-97] Sarvar Patel, "Number Theoretic Attacks On Secure Password Schemes", 1997 IEEE Symposium on Security and Privacy, pp.236-247, May 1997.

[Paulson-97a] Lawrence C.Paulson, "Proving Properties of Security Protocols by Induction", 10th IEEE Computer Security Foundations Workshop, pp.70-83, Jun.1997.

[Paulson-97b] Lawrence C.Paulson, "Mechanized Proofs for a Recursive Authentication Protocol", 10th IEEE Computer Security Foundations Workshop, pp.84-91, Jun.1997.

[Peterson-61] W.Wesley Peterson,"Error-Correcting Codes", MIT Press,1961.

[Peterson-72] W.Wesley Peterson and E.J.Weldon,"Error-Correcting Codes", 2nd edition, MIT Press,1972.

[Pfitzmann-97] Andreas Pfitzmann, Birgit Pfitzmann, Matthias Schunter and Michael Waidner, "Trusting Mobile User Devices and Security Modules", IEEE Computer, pp.61-68, Feb.1997.

[Ragget-95] Dave Raggett,"Mediated Digest Authentication", Internet Draft draft-ietf-http-mda-00, Mar.1995.

[Reiter-97] Michel K.Reiter and Stuart G.Stubblebine, "Toward Acceptable Metrics of Authentication", 1997 IEEE Symposium on Security and Privacy, pp.10-20, May 1997.

[RFC1004-87] D.L.Mills,"A Distributed-Protocol Authentication Scheme",Apr.1987.

[RFC1334-92] B.Lloyd and W.Simpson,"PPP Authentication Protocols", Oct.1992.

[RFC1352-92]J.Galvin,K.McCloghrie and J.Davin," SNMP Security Protocols",Jul.1992.

[RFC1446-93]J.Galvin and K.McCloghrie,"Security Protocols for version 2 of the Simple Network Management Protocol(SNMPv2)", Apr.1993.

[RFC1507-93] C.Kaufman, "Distributed Authentication Security Service(DASS)", Sep.1993.

[RFC1510-93] J.Kohl and C.Neuman,"The Kerberos Network Authentication Service

(V5)",Sep.1993.

[RFC1688-94]W.Simpson," IPng Mobility Consideration",Aug.1994.

[RFC1704-94] N.Haller and R.Atkinson,"On Internet Authentication",Oct.1994.

[RFC1826-95] R.Atkinson,"IP Authentication Header", Aug.1995.

[Roscoe-96] A.W.Roscoe,"Intensional specification of security protocol",9th IEEE Computer Security Foundations Workshop, pp.28-38,June 10-12,1996.

[Samfat-94] D.Samfat and R.Molva "A Method Providing Identity Privacy to Mobile Users during Authentication" Workshop on Mobile Computing Systems and Applications,1994.

[Schneider-97] Steve Schneider, "Verifying authentication protocols with CSP", 10th IEEE Computer Security Foundations Workshop, pp.3-17, Jun.1997.

[Schneier-94] Bruce Schneier,"Applied Cryptography",John Wiley & Sons,Inc.,1994.

[Schuba-97] Christph L.Schuba, Ivan V.Krsul, Markus G.Kuhn, Eugene H.Spafford, Aurobindo Sundaram and Diego Zamboni, "Analysis of a Denial of Service Attack on TCP", 1997 IEEE Symposium on Security and Privacy, pp.208-223, May 1997.

[Sinclaie-96] Jane Sinclair,"Action Systems for Security Specification",9th IEEE Computer

Security Foundations Workshop, pp.102-113,June 10-12,1996.

[Steiner-88] J.G.Steiner, B.C.Neuman, J.I.Schiller, "Kerberos : An Authentication Service for Open Network Systems"(Version 4), Proc.of the Winter USENIX Conference, pp.191-202, Feb.1988.

[Syverson-96] Paul Syverson,"Limitation on Design Principles for Public Key Protocols",1996 IEEE Symposium on Security and Privacy,pp.62-72,May 6-8,1996.

[Wilkers-95] Joseph E.Wilkes, "Privacy and Authentication Needs of PCS", IEEE Personal Communications, Vol.2,No.4, pp.11-15, Aug.1995.

[Williams-96] Veronica A.Williams ,"Wireless Computing Primer - A Comprehensive Guide to Wireless and Mobile Computing", M&T Books, 1996.

[Woo-92] T.Y.C.Woo and S.S.Lam,"Authentication for distributed systems", Computer, vol.25,no.1, pp.39-52, Jan.1992.

[Woo-94] T.Y.C.Woo and S.S.Lam,"A lesson on authentication protocol design", Operating Systems Review,vol.28,no.3,pp.24-37,Jul.1994.

[Young-97] Adam Young and Moti Yung, "Denial Password Snatching: On the Possibility of Evasive Electronic Espionage", 1997 IEEE Symposium on Security and Privacy, pp.224-235, May 1997.

[Zhou-96] Jianying Zhou and Dieter Gollmann,"A Fair Non-repudiation Protocol",1996 IEEE Symposium on Security and Privacy,pp.55-61,May 6-8,1996.

[日経BP-93] 最前線レポート モービル・コンピューティング, 日経BP社(1993).

[水野-96] 水野忠則:離陸するモーバイルコンピューティング, モーバイルコンピューティングシンポジウム, 情報処理学会モーバイルコンピューティング研究グループ, pp.1-6,1996.

関連発表論文

1)和田雄次、田窪昭夫、溝口徹夫、"順編成ファイルのジャンプ・サーチ",信学技報EC80-1、電子通信学会電子計算機研究会,Apr.1980.

2)田窪昭夫、竹沢明、塚本久雄、飯塚信雄、"MELCOM80OCRオフィスコンピュータ",三菱電機技報Vol.56,No.2,Feb.1982.

3)赤桐行昌、田窪昭夫、柴田信之、江村弘、渡部重彦、"三菱パーソナルコンピュータMULTI8",三菱電機技報Vol.57,No.12,Deec.1983.

4)沢井喜彦、田窪昭夫、堂坂辰、青井伸、中川路哲男、"パソコンを中心としたクライアントサーバコンピューティングの今後"、三菱電機技報Vol.65,No.12,Dec.1991.

5)水野、田窪:モーバイルコンピューテイング、2010年マルチメディア通信と高速・知能・分散協調コンピューティングシンポジウム Vol.94,No.7,1994.

6)水野、田窪:モーバイルコンピューテイングシステムモデルの提案、情報処理学会研究報告 95-DPS-68,95-GW-9, Vol.95,No.13,1995.

7)田窪、石川、水野:逐次捕捉型モーバイルコンピューテイングシス環境におけるセキュリティ方式の考察、情報処理学会研究報告 95-DPS-71, Vol.95,No.61, pp.79-84,1995.

8)水野、田窪:モーバイルコンピューティング、情報処理 Vol.36,No.9,pp.822-826, 1995.

9)石川、田窪、水野:モーバイルコンピュータ環境におけるユーザ認証方式, 第 52 回情報処理学会全国大会講演論文集、pp.1-81-82, 1996.

10)石川, 田窪, 渡辺, 水野:モーバイルコンピューティング環境におけるユーザ認証方式とその評価,情報処理学会研究報告, 情処研報 Vol.96, No.MBL-2, pp.13-18,1996.

11)田窪、石川、渡辺、曽我、水野:"モーバイルコンピューティング環境における認証方式の提案"、静岡大学大学院電子科学研究科、研究報告第18号、1996.

12)A.Takubo,M.Ishikawa,T.Watanabe and T.Mizuno, "Authentication Protocol for Mobile Computing Environment", The IEEE 11th International Conference on Information Networking, 1997.

13)石川、田窪、渡辺、水野:モーバイルコンピューティング環境におけるユーザ認証とセキュリティシミュレータ、第 54 回情報処理学会全国大会論文集、pp.3-537, 1997.

14)石川、田窪、渡辺、水野:"モーバイルコンピューティング環境におけるセキュリティシミュレータについて", マルチメディア,分散, ｸﾞﾙｰﾌﾟｳｪｱとモーバイル(DiCoMo)ワークショップ論文募集,pp.599-604,Jul.1997.

15)A.Takubo,M.Ishikawa,T.Watanabe,M.Soga and T.Mizuno,"User Authentication in Mobile Computing Environment",IEICE Trans.on Fundamentals of Electronics, Communications and Computer Sciences,Vol.E80-A,No.7,pp.1288-1298,Jul.1997.

16)A.Takubo,M.Ishikawa,T.Watanabe and T.Mizuno, “Security Simulator in Mobile Computing Environment”, The 12th International Conference on Information Networking, Jan.1998.

17)田窪、石川、渡辺、水野:"移動計算環境におけるセキュリティ・シミュレータの実装"、静岡大学大学院電子科学研究科、研究報告第19号、1997.(印刷中)