知能と複雑系 117－6
(1999. 7. 22)

なりすましに対する不正侵入検知システム (IDS-M)

岡本 忠士† 白石 善明† 大家 隆弘‡ 森井 昌克†

† 徳島大学 工学部 知能情報工学科 ‡ 徳島大学 工学部 電気電子工学科
〒770-8506 徳島市南常三島町 2-1
E-mail: {tadasi,zenmei,morii}@is.tokushima-u.ac.jp E-mail: alex@ee.tokushima-u.ac.jp

あらまし 不正侵入に対する防御システム (侵入者検知システム，IDS) がいくつか開発されている．これらのシステムは既知の不正侵入方法との照合を行うことで，侵入者からネットワークを防御する．しかしながら，このような照合に基づく方式では，なりすましなど未知の不正侵入には対応出来ない．また，これらは外部からの侵入検出を目的としているので，ネットワーク内部の不正行為は検出できない．

本研究では内部ネットワークに流れるパケットを監視することにより，ネットワーク内外の不正ユーザが正当なユーザになりすましているのを検出するシステムを提案する．なりすまし検出は，ユーザが入力した文字の生起確率と，入力速度を調べることにより実現している．本稿ではシステムの概要を述べると共に，実際のネットワーク上での実験結果とその考察を与える．

キーワード 不正アクセス，侵入者検知システム，なりすまし，ユーザ識別

Intruder Detection System for Masquarade(IDS-M)

Tadashi OKAMOTO† Yoshiaki SHIRAISHI† Takahiro OIE‡ Masakatu MORII†

† Department of Information Science & Intelligent Systems, Tokushima University ‡ Department of Electrical & Electronic Engineering, Tokushima University
2-1 Minamijyosanjima, Tokushima 770-8506, Japan
E-mail: {tadasi,zenmei,morii}@is.tokushima-u.ac.jp E-mail: alex@ee.tokushima-u.ac.jp

Abstract There are some Intruder Detection Systems(IDS's) to defend networks and systems against the illegal access. Most systems check the current access with the known illegal access pattern in the database. However it can not cope with unknown illegal access, i.e. masquarade and so forth. Moreover, it can not be detected that illegal access through inner network, because the purpose of these IDS's is to detect the attacker from outer network.

In this paper, we propose a new IDS called IDS-M(IDS for Masquarade) which detects that the attacker in the attacker in the inner and outer networks masquarade as the right user by monitoring the packets from the outer network and through the inner network. The occurrence probability of the user's inputted characters, and the user's input speed are used in this method. Furthermore, we give the experimental result in our network and some considerations.

key words Illegal Access, Masquerade, Intruder Detection System, Masquarade Detection

1 はじめに

コンピュータネットワークは学術ネットワークから企業や，一般家庭にいたるまで急速に広がりつつある．一方でそれらネットワークを利用して外部からアクセスすることを許可されていないマシンに不正アクセスする侵入者の存在や，内部の関係者による犯罪が問題となっている．ここでは，外部からの不正アクセスユーザと，内部のなりすましをしているユーザをあわせて侵入者と呼ぶことにする．

侵入者は内部の正当なユーザになりすまして個人や企業，団体等のコンピュータシステムに侵入し，情報の搾取・破壊や運用妨害を行い，多大な被害をもたらす場合もある．最近，この不正侵入行為に対する防御システム(侵入者検知システム，IntruderDetection System:IDS)がいくつか開発されている [1] [2]．それらのシステムは，基本的に既知の侵入方法，およびセキュリティホールをデータベース化しておき，モニタリングしたパケットをそのデータベース上の侵入方法等と照合することで不正アクセスを検知する．

このように従来の不正アクセスを事前に検知し，排除するという方法ではなく，本研究では不正アクセスが行われた場合，できるだけ早急に検知し，被害を最小限に押さえる不正アクセス検知システムの開発を行う．具体的には外部ネットワークから内部ネットワークへのパケット，並びにネットワーク内を流れるパケットをモニタリングすることによって，不正アクセスを含めて，正当なユーザであるかどうかを判定する．

すなわち，内部ユーザの特徴を詳細に記述したデータベースをユーザのアクセス履歴から作成し，ユーザの個人的な特徴を抽出し，その特徴から正規のユーザ，すなわちなりすましが行われていないユーザであるか否かを判定する．なお，ユーザの個人的特徴は逐次的に更新される．

本稿では 2 章でなりすましに対する不正侵入検知システム (IDS for Masquarade:IDS-M) の概要を述べ，3 章では提案したユーザ判定を行う方法と，実際のネットワーク上でシステムを運用した実験結果を与え，それらについて考察を与える．

2 システム概要

2.1 IDS-M サーバの機能

監査対象となるホストへのパケットを採取し，なりすまし検出を行うために，同一ネットワーク内に本システムのサーバ (IDS-M サーバ) を配置する．その一例を図 1に示す．この例では 4 台のホストへのパケットを監視するために，IDS-M サーバを 4 台のホストと同一ネットワーク上に配置している．インターネットを介してこれらのホストに送信されたデータと内部ネットワークを流れるデータはすべてこのサーバでモニタリングすることができる．

実験システムでは監査対象となる UNIX サーバ 4 台が接続しているネットワーク内に本システムのサーバ

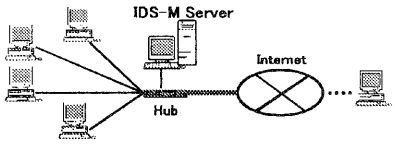

図 1: ネットワーク構成図 (例)

(CPU : SuperSPARC, memory : 64Mbyte, OS : Sun OS4.1.4) を配置した．今回の実験ではログイン先のホスト名までは考慮しないように UNIX サーバはすべて NIS でユーザ情報を一括管理しており，ホームディレクトリは NFS でマウントされているという環境である．

次に IDS-M サーバ内のシステム構成を図 2に示す．IDS-M はユーザ判定部とパケット取得部から構成される．まずパケット取得部ではネットワーク中のパケットが nit(Network InterfaceTap) と呼ばれる擬似デバイスに渡され，フィルタリングプログラムによって，rlogin と telnet のパケットを選択し，クライアント側から送信されるデータを採取する．ユーザ判定部のデータベースには，それぞれの検出プログラムによって作成された監査データが保存されている．監査データとは，検出プログラムによって異なる方法で構成された正当なユーザの特徴データである．パケット取得部より受け取ったデータはそれぞれの検出プログラムに渡され，データベース内の監査データを元に侵入者の検出を行なう．

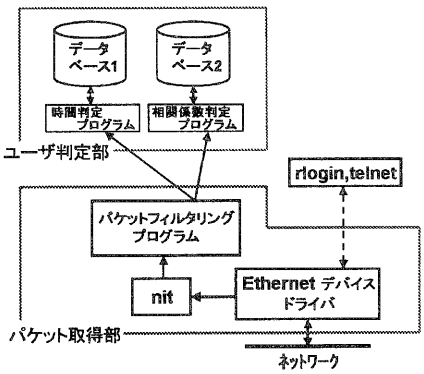

図 2: システム構成図

2.2 パケット取得部

SunOS では nit と呼ばれる擬似デバイスが提供されており，このデバイスを利用することで他のホストから送信されるパケットを採取できる．本システムでは nit を利用して，遠隔ログインを行なうアプリケーションである rlogin，telnet のパケットをフィルタリングする．rlogin，telnet プロトコルは単にデータを転送するだけであり，

またサーバ側から送信されるデータ量はクライアント側から送信されるデータ量に比べて多いため，データの採取はクライアント側から送信されるデータを重要視し，サーバから送信されるデータはユーザ名を取得するときのみ利用する．

2.2.1 ユーザ名の取得方法

本システムでは，侵入者がすでに何らかの方法でユーザ名とそのパスワードを取得している，もしくは，".rhosts" ファイルに書かれている情報によって自由にログインできることを前提としている．IDS-M ではユーザ名によってデータベースを構築し，照合を行うため，ユーザ名も採取する必要がある．

rlogin の場合には最初のパケット中に含まれるサーバ側のユーザ名を採取しておく．telnet でのユーザ名の採取はサーバ側から送信される " login: " の文字列を検出し，認証の完了はサーバ側から送信される "Last login:" の文字列† を検出することで行なった．

2.3 ユーザ判定部

IDS-M サーバの擬似デバイスがネットワークからパケットを取得し，そのパケットからパケット取得部が検出する情報は，パケットが到着した時間と，ユーザが入力したキーの情報のみである．

そこで本研究では以下の (1)，(2) のようにパケットの到着時刻と，ユーザがタイプしたキーの情報を利用する．

(1) 連続する 2 つのパケットの到着時刻の差からユーザがキーをタイプする間隔を求める．さらにこの中でユーザの特徴がもっと現れやすいと思われる，コマンドを入力中のタイピング間隔を抽出する．

(2) タイプしたキーの情報とはユーザが入力した文字キーのアスキーコードである．このアスキーコードの系列からユーザの特徴を抽出する．このためアスキーコードの系列を 2 重マルコフ過程として，2 文字間の結合確率を考える．

(2) において，2 重マルコフ過程とした理由と，コマンド単位の生起確率でなく，2 文字単位にしたのは以下の理由からである．

- 1 文字は 2 文字間の結合確率に含まれるデータであり，n 以上 $(n > 2)$ の n 重マルコフ過程では，結合確率のテーブルが n 次元となり，計算量が増加するため．
- ユーザがキーを入力してもネットワークの環境や回線状況によってはサーバに届かなかったり，届いても入力した順番通りに届かない場合もある．さらに，使用するリモートログインアプリケーションによっては完全なコマンド列を入力しなくても補完してくれる機能などがある．このため本研究の目標とするシステムはネットワークの監視によってユーザを判定することであるが，ネットワークの監視だけでコマンド列を採取するのは困難であるので，2 文字単位とした．

また，(2) においては対象とするアスキーコードは，NVT ASCII で指定されている 7bit キャラクタの 128 文字のうち，文字に対応するアスキーコード 33 番から 126 番までの 94 文字とした．この理由は以下の 2 つである．

- 128 番以降は主に日本語で使用されるが，端末上のコマンド入力ではあまり使用されない文字である．
- 0 番から 32 番，127 番のアスキーコードは，コントロールコードと呼ばれ，Enter キーやスペースキーなど，文字以外の入力に対応している．これらの文字は一般的によく使用されるため，他の文字の情報量がこれらのコントロールコードによって縮小されてしまう．

(1)，(2) の方法とも，以下の監査データと，ユーザデータとよばれる 2 つのデータを用意し，そのデータ間の差異を求めることで判定を行う．

ユーザデータ： あるユーザがログインすると初めて作成されるデータ．この作成方法も判定プログラムによって異なる．ログイン中にユーザが文字を入力するたびに逐次更新されていくが，ユーザがログアウトすると，このデータは消去される．

監査データ： データベースに保存されている，検査対象となるネットワーク内部に登録されている各ユーザ（正当なユーザ）ごとに存在するデータ．このデータには判定プログラムによって構成方法の違う正当なユーザの特徴が格納されており，正当なユーザと判定されたユーザのログイン情報（ユーザデータ）によって逐次更新されていく．

3 ユーザ判定手法および実験結果

3.1 2 重マルコフ過程による判定

3.1.1 方法

ユーザが入力したキーの情報からユーザの特徴を表す方法として，タイプされた前後 2 文字間の結合確率を利用する．

ログイン中のユーザから求められる結合確率のテーブル y'_{c_1,c_2} は全体の文字数を k とすると式 (1)，式 (2) で表される．c_1 は直前にタイプされた文字のアスキーコードで，c_2 は現在タイプされた文字のアスキーコードである．ここで，対象とする文字はアスキーコード 33 番から 126 番までの 94 文字とする．よって $k = 94$ である．また，記号の都合上アスキーコード 33 番を 0 に対応させ，$0 \leq c_1, c_2 \leq 93$ とする．

†今回は実験では SunOS4.1.4 の場合のみ扱った．他の OS のログイン完了メッセージに対応させることにより，複数の OS にも本システムは適用できる．

$$y'_{c_1,c_2} = \Pr(c_1, c_2) \tag{1}$$

$$\sum_{c_1=0}^{k-1} \sum_{c_2=0}^{k-1} y'_{c_1,c_2} = 1 \tag{2}$$

同様に，監査データの結合確率のテーブルx'_{c_1,c_2}も，以下の式(3)，式(4)によって求める．

$$x'_{c_1,c_2} = \Pr(c_1, c_2) \tag{3}$$

$$\sum_{c_1=0}^{k-1} \sum_{c_2=0}^{k-1} x'_{c_1,c_2} = 1 \tag{4}$$

この方法で利用する監査データとユーザデータについて述べる．ユーザデータには，結合確率のテーブルと総入力文字数が格納される．この結合確率のテーブルは1文字入力されるたびに更新する．監査データはこれまでのログインで更新されてきた結合確率のテーブルと，これまでのすべてのログインで入力した文字数の総和が含まれている．監査データと，ユーザデータの結合確率のテーブルの差異を数値的に求めるために，両データの相関係数を求める．

この相関係数は次のようにして求められる．

まず，監査データと，ユーザデータの結合確率のテーブルより，それぞれの特徴ベクトルを以下の式(5)，式(6)によって求める．

$$\begin{aligned} \mathbf{y} &= (y_0, y_1, y_2, \cdots, y_{m-1}) \\ &\stackrel{def}{=} (y'_{0,0}, y'_{0,1}, y'_{0,2}, \cdots, y'_{k-1,k-1}) \end{aligned} \tag{5}$$

$$\begin{aligned} \mathbf{x} &= (x_0, x_1, x_2, \cdots, x_{m-1}) \\ &\stackrel{def}{=} (x'_{0,0}, x'_{0,1}, x'_{0,2}, \cdots, x'_{k-1,k-1}) \end{aligned} \tag{6}$$

これらの特徴ベクトルは結合確率から以下のように生成される．

- m は結合確率の 2 次元テーブルの要素数である．$94 \times 94 = 8836$ より，$m = 8836$ である．また，結合確率のテーブルとベクトルの要素との対応は，$y'_{c_1,c_2} = y_{(c_1 \times 94 + c_2)}$，$x'_{c_1,c_2} = x_{(c_1 \times 94 + c_2)}$ で与えられる．
- 各成分はある 2 文字が連続して生起する確率である．

これらより，相関係数 $u'(\mathbf{x},\mathbf{y})$ は以下の式(7)によって求められる．

$$u'(\mathbf{x},\mathbf{y}) = \frac{u(\mathbf{x},\mathbf{y})}{\|\mathbf{x}\|\|\mathbf{y}\|} \tag{7}$$

ここで，

$$\|\mathbf{x}\| = \sqrt{\sum_{i=0}^{m-1} {x_i}^2} \quad , \quad \|\mathbf{y}\| = \sqrt{\sum_{i=0}^{m-1} {y_i}^2} \tag{8}$$

$$u(\mathbf{x},\mathbf{y}) = \sum_{i=0}^{m-1} x_i y_i \tag{9}$$

式(7)で表される相関係数の意味について考える．内積は監査データとユーザデータの特徴ベクトル x と y で，同じ 2 文字の結合確率 $(x_{n_1}, y_{n_2}, n_1 = n_2)$ が，高いほど大きな値をとる．つまり，両データで，よく入力される文字列の個数が多いほど，またその文字列を構成する文字数が多いほど大きな値をとる．

また，内積の最大値は特徴ベクトル x と y において同じ添え字をもつすべての要素が等しく，$x_{n_1} = y_{n_1}$ となるときであり，このとき式(9)，式(8)より，

$$u(\mathbf{x},\mathbf{y}) = \|\mathbf{x}\|^2 = \|\mathbf{y}\|^2$$

$$\|\mathbf{x}\|\|\mathbf{y}\| = \|\mathbf{x}\|^2 = \|\mathbf{y}\|^2$$

となるので，相関係数の最大値は 1 となる．このときログイン中のユーザは，監査データ（正当なユーザ）と同じ文字を同じ順序で入力していることになる．

逆に内積が最小とは，監査データとユーザデータの 2 つの特徴ベクトルにおいて，同じ添え字を持つすべての要素で，どちらかが 0 であった場合であり，このとき内積は 0 になる．よって，相関係数の最小値は 0 である．このとき，ログイン中のユーザは監査データ（正当なユーザ）で入力したことのない順序で文字を入力していることになる．

以上より，相関係数の範囲は $0 \leq u'(\mathbf{x},\mathbf{y}) \leq 1$ となる．この相関係数 $u'(\mathbf{x},\mathbf{y})$ を用いてユーザを判定する．判定は，ある閾値 Th をもうけ，以下のように判定する．

$u'(\mathbf{x},\mathbf{y}) \leq Th$ ：なりすましのユーザ
$u'(\mathbf{x},\mathbf{y}) \geq Th$ ：正当なユーザ

3.1.2 ユーザの判定実験

実験期間は 1999 年 4 月 25 日から 6 月 20 日までの約 2ヶ月間である．以下の手順で実験を行った．

実験 1： 9 名の正当なユーザのデータにより，閾値 Th を決定する．

実験 2-1： 実験 1 で求めた閾値 Th を用いて，正当なユーザがログインしたときのユーザを判定する．

実験 2-2： 実験 1 で求めた閾値 Th を用いて，なりすましたユーザがログインしたときのユーザを判定する．

閾値 Th の決定は以下のようにして行った．

- 9 名のユーザの数回のログインでの相関係数を求める．
- 9 名全員分の相関係数から，50%信頼区間の下限値を求め、その値を閾値 Th とする．

9 名 (user1,user2,$\cdots$,user9) のユーザの数回のログインにおいて求められたすべての相関係数の平均と標準偏差をユーザごとに求める．その結果を表 1に示す．実験ではユーザがログインしてから 40 文字入力するごとに相関係数を求める．例えば 5 回のログインで，毎回 400 文字入力した場合は 50 個の相関係数が得られることになる．最後に、各ユーザごとに求められたすべての相関係数を用いて，9 名全員分の平均と標準偏差を求めた．また，表 1中のデータ数とは得られた相関係数の個数であり，all の行が全体の平均と標準偏差である．

この all の行の値から，50%信頼限界の下限値を求め閾値 Th とする．相関係数の平均を $\overline{u'}$，標準偏差を S_2，また，$\alpha = 0.674$ とすると，式(10) と表1より $Th = 0.2422$ を得る．

$$Th = \overline{u'} - \alpha S_2 \tag{10}$$

ここで 50%としたのは，それよりも信頼区間を広げるとなりすましたユーザを正当なユーザと判定する可能性が高くなるためでる．逆に狭くすると，正当なユーザをなりすましたユーザと判定してしまうからである．

表 1: (実験 1) 正当なユーザがログインしたときの相関係数の平均と標準偏差

ユーザ	平均	標準偏差	データ数
user1	0.5987	0.3518	254
user2	0.1688	0.1076	108
user3	0.3949	0.2892	91
user4	0.2494	0.0888	66
user5	0.2784	0.2398	117
user6	0.5697	0.3265	158
user7	0.4976	0.3779	90
user8	0.5004	0.1668	143
user9	0.4681	0.2272	120
all	0.4514	0.3104	1147

実験 2-1 では，この閾値 Th の検証を行った．ある不正なユーザ (user11) が実験 1 の全てのユーザになりすましてログインしたときの判定結果を表 2に示す．このときなりすましたユーザは約 360 個の文字を入力した．40 文字ごとに相関係数の計算を行ったので，データ数は，9 回 $(360/40 = 9)$ である．

実験 2-2 では，これまでとは異なる，正当なユーザ (user10) に対して閾値の検証を行った．実験を開始してからこのユーザが，数回ログインし，その直後から，実験 2-1 と同じなりすましたユーザ (user11) が数回ログインした．この正当なユーザがログインした間と，なりすましたユーザがログインした間の 2 つの期間中の判定結果を表 3に示す．

表 2: (実験 2-1) なりすましたユーザ (user11) がログインしたときの相関係数の平均と標準偏差，および閾値 Th による判定成功率

ユーザ	平均	標準偏差	データ数	成功回数
user1	0.0550	0.0176	9	9
user2	0.1795	0.0448	9	9
user3	0.1233	0.0093	9	9
user4	0.0792	0.0156	9	9
user5	0.0726	0.0183	9	9
user6	0.0469	0.0041	9	9
user7	0.0981	0.0179	9	9
user8	0.5147	0.1099	9	1
user9	0.3356	0.0758	9	1
all	0.1672	0.1567	81	65

3.2 タイピング間隔による判定

3.2.1 方法

本研究の実験で使用した IDS-M サーバは，1970 年から経過した時間を 10msec 単位で計測している．この時刻を文字を取得するたびにファイルに記録しておく．

通常は，1 パケットに 1 文字が含まれているので，文字の入力時間の差はパケットの到着時間の差とみなすことができる．n 個のパケット $p_1, p_2, \cdots, p_n$ の到着時刻が $t_1, t_2, \cdots, t_3$ とすると，タイピング間隔は以下のように計算できる．

$$t_2 - t_1, t_3 - t_2, \cdots, t_n - t_{n-1}$$

ログイン中のタイピング間隔の平均値と，監査データに保存されている平均値との差を求め、t 検定によりその有意性を検討することでユーザの判定を試みる．もし有意差がなければ正当なユーザと判定する．また，監査データには，正当なユーザの過去のログイン時のタイピング間隔が全て記録されている．

まず，現在ログインしているユーザのユーザデータから t 検定に必要な平均値と，分散を以下のようにして求める．

ユーザの真のタイピング間隔が知りたいので，コマンド系列を入力するとき以外のタイピング間隔を除くために，以下の条件を設ける．

条件 1： Enter キーを押されてから次のコマンドの先頭文字までの時間は考慮しない．

条件 2： 10 秒以上の時間は考慮しない．

条件 1 は，コマンドから次のコマンドを入力するまでのユーザの思考時間を除くためであり，条件 2 では，ユーザがログイン中に一時的に入力を中断したときの時間を除くためである．さらにコマンド入力時以外のタイピング間隔を除くため，条件 1，条件 2 を満たすタイピング間隔から，その平均 $\overline{t_y}$ と分散 s_y^2 を求め、99%信頼限界を求める．さらに条件 3 を追加する．

表 3: (実験 2-2) 正当なユーザ (user10) となりすましたユーザ (user11) がログインした場合の閾値 Th による判定成功率

ユーザ	正当なユーザのログイン	なりすましたユーザのログイン
正当なユーザと判定した回数	123(62.4%)	1(1.1%)
なりすましユーザと判定した回数	74(37.6%)	92(98.9%)
全体の判定回数	197	93
判定成功率	62.4%	98.9%

条件 3： 条件 1,2 を満たすタイピング間隔の 99%信頼限界の上限値よりも大きなタイピング間隔は考慮しない

ログイン開始から N 文字入力されるまでに抽出された n_y 個のタイピング間隔 $t_{y,i}(1 \leq i \leq n_y)$ より，その平均 $\overline{t_y}$ と分散 s_y^2 をそれぞれ以下の式 (11)，式 (12) で計算する．

$$\overline{t_y} = \sum_{i=1}^{n_y} t_{y,i} \tag{11}$$

$$s_y^2 = \frac{\sum_{i=1}^{n_y}(t_{y,i} - \overline{t_y})^2}{n_y} \tag{12}$$

この平均 $\overline{t_y}$ と分散 s_y^2 から，99%信頼限界の上限値 w を式 (13) によって求める．

$$w = \overline{t_y} + 2.576\sqrt{s_y^2} \tag{13}$$

条件 3 より，w 以下となるタイピング間隔と，その個数を求め、再び平均と分散を求める．

(例 1)

あるユーザにおいて条件 1，2 をみたすタイピング間隔のグラフを図 3に示す．このユーザの最初に求められた平均は 0.5624[sec]，標準偏差は 1.0699[sec] で，そこから求められる 99%の信頼限界上限は 3.3185[sec] であった．図 3の矢印で示した値がこの時間より大きな値である．条件 3 よりこの上限値を上回るタイピング間隔を除いて再び求めた平均は 0.4093[sec]，標準偏差は 0.5877[sec] であった．

(例終わり)

次に，t 検定による判定方法について述べる．監査データについてはすでに，以上と同様な方法で計算された平均と分散が求められているものとする．監査データから求められる平均を $\overline{t_x}$，分散を s_x^2，タイピング間隔のデータ数を n_x とし，ログイン中のデータであるユーザデータから求められる平均を $\overline{t_y}$，分散を s_y^2，タイピング間隔のデータ数を n_y とする．t 値 t_{xy} を以下の式 (14) によって求める．

$$t_{xy} = \frac{|\overline{t_x} - \overline{t_y}|}{\sqrt{\frac{s_x^2 n_x + s_y^2 n_y}{n_x + n_y - 2}\left(\frac{1}{n_x} + \frac{1}{n_y}\right)}} \tag{14}$$

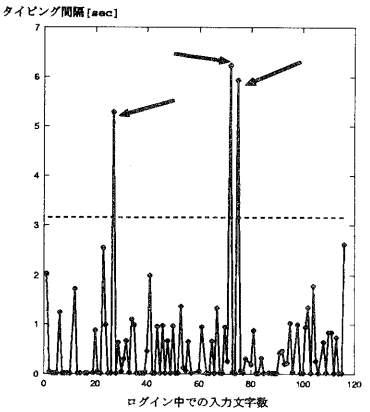

図 3: 条件 1，2 を満たすあるユーザのタイピング間隔

ここで，検定を行うための帰無仮説 (H_0) と，対立仮説 (H_1) を以下のようにする．

帰無仮説 (H_0)： 監査データとユーザデータのタイピング間隔 $\overline{t_x}$ と $\overline{t_y}$ に有意差は無く，両データのユーザは同一ユーザである．(正当なユーザと判定)

対立仮説 (H_1)： 監査データとユーザデータのタイピング間隔 $\overline{t_x}$ と $\overline{t_y}$ に有意差があり，両データのユーザは異なるユーザである．(なりすましたユーザと判定)

つまり，H_0 が支持されればログイン中のユーザは正当なユーザであり，棄却されればなりすましのユーザであると判定できる．

本研究では，有意水準 5%で t 検定を行い第 1 種の誤りが起こる確率と第 2 種の誤りが起こる確率を求め、実験の検証を行った．この場合の第 1 種の誤りと，第 2 種の誤りは以下のようになる．

第 1 種の誤り： 正当なユーザをなりすましたユーザと判定してしまう誤り

第 2 種の誤り： なりすましたユーザを正当なユーザと判定してしまう誤り

3.2.2 ユーザの判定実験

実験期間は1999年4月25日から6月20日までの約2ヶ月間である．以下の手順で判定を行い，実験1，2と同じユーザ10名 (user1,user2,$\cdots$,user10) について，その実験結果を示す．

実験3： L 回目 ($L > 2$) に正当なユーザがログインした場合を判定

実験4： L 回目 ($L > 2$) になりすましたユーザがログインした場合を判定

実験3は，第1種の誤りが起こる確率を求めるためであり，実験4は第2種の誤りが起こる確率を求めるための実験である．実験4で正当なユーザとは異なるユーザとして，タイピング間隔が短かく，ブラインドタッチをしていると思われるユーザ (userX とする) のデータを利用した．

実験3の結果を表5に示す．これは正当なユーザが6回ログインしたときの判定結果である．

次に実験4の結果を表6に示す．これはuserXが，各ユーザになりすました場合の判定結果である．例えば表6で，user2の6回のところの“正”は，userXが6回目にuser2になりすましてログインしたことを想定し，user2が5回続けてログインしたときに構成された監査データと，userXのある1回分のログインのデータを6回目のログインのユーザデータとして判定した結果，userXがなりすましたユーザと判定されたことを示している．

以上の実験3と実験4の集計結果を表4に示す．これらより，第1種，第2種の誤りが起きた確率は以下のようになる．

第1種の誤りが起きた確率 = 12%(6/50=0.12)
第2種の誤りが起きた確率 = 14%(7/50=0.14)

表4: 実験3，4での判定結果の集計

	正当なユーザと判定した回数	なりすましユーザと判定した回数	全体の判定回数
実験3	44(88%)	6(12%)	50
実験4	7(14%)	43(86%)	50

また，実験3，4で最後のログインを判定したときの各ユーザの監査データと，userXのユーザデータを表7に示す．

3.3 考察

表2において，全体の判定成功率は 65/81 = 0.802(80.2%) という高い成功確率を得ているが，user8とuser9は，11%という低い成功率であった．さらに，user10においても正当なユーザに対する判定成功率は62.4%と低い．しかし，実験3，4では，user8，user9，user10とも高い判定成功率を得ている．

また，実験2-2(表3)では，正当なユーザ，なりすましユーザともその判定成功率は50%を超えており，この閾値 Th は有効であったと言える．

実験3，4で第1種，第2種の誤りが起きる確率を求めた結果，第1種の誤りが起きた確率，第2種の誤りが起きた確率ともに小さく，全体としては良い結果が出た．しかし，実験4でuser1はなりすましたユーザを正当なユーザと誤判定しており，良い結果が出ていない．これはユーザの正確なタイピング間隔をもとめられていないことや，1パケットに複数の文字が含まれる場合があるからである．ネットワークの環境や変化する回線状況によってはパケットの到着が遅れることがある．通常1パケットに1文字が含まれるので，3つのパケット p_1, p_2, p_3 の到着時刻が，それぞれ t_1, t_2, t_3 であるとき，$t_1 \leq t_2 \leq t_3$ となるが，到着が遅れる場合，1パケットに複数の文字が含まれる場合がある．つまり，$t_1 = t_2 = t_3$ となる．このような場合では正確にタイピング間隔を知

表5: (実験3) 正当なユーザがログインした場合の判定結果 (正：正当なユーザと判定，誤：なりすましユーザと判定)

ユーザ	2回	3回	4回	5回	6回
user1	正	正	誤	正	正
user2	正	正	正	正	正
user3	正	誤	誤	正	正
user4	正	正	正	正	正
user5	正	正	正	正	正
user6	正	誤	正	正	正
user7	正	正	誤	正	正
user8	正	正	正	正	正
user9	正	誤	正	正	正
user10	正	正	正	正	正

表6: (実験4) なりすましたユーザがログインした場合の判定結果 (誤：正当なユーザと判定，正：なりすましユーザと判定)

ユーザ	2回	3回	4回	5回	6回
user1	誤	誤	誤	誤	誤
user2	正	正	正	正	正
user3	正	正	正	正	正
user4	正	正	正	誤	正
user5	正	正	正	正	正
user6	正	正	正	正	正
user7	正	正	正	誤	正
user8	正	正	正	正	正
user9	正	正	正	正	正
user10	正	正	正	正	正

表 7: (実験 3，4) 各ユーザの監査データに含まれるデータと userX のデータ

ユーザ	全データ数	信頼区間 [sec]	タイピング間隔の平均 [sec]	タイピング間隔の偏差 [sec]	全条件を満たすデータ数
user1	12240	1.8488	0.1937	0.3352	11088
user2	2947	3.3210	0.2591	0.5519	2677
user3	1019	2.7105	0.3539	0.5378	918
user4	851	2.5398	0.2535	0.4132	757
user5	3582	2.9229	0.3578	0.4658	3274
user6	905	1.8071	0.1082	0.2849	880
user7	73	2.5194	0.4147	0.5423	59
user8	3488	4.9005	0.7036	1.1120	3052
user9	3537	3.0894	0.3687	0.5053	3080
user10	133	3.2612	0.5022	0.6294	126
userX	126	1.5607	0.1897	0.3168	111

ることは出来ない．

しかし，実験 2-1 ではなりすましユーザの判定成功率は 100%となっている．

タイピング間隔による判定が利用する入力時間と，相関係数による判定で利用する 2 文字間の結合確率は，互いに独立したデータである．このため，上記のようにどちらかの方法を利用することによってユーザの判定成功率を高めることができる．これらを表 8に示す．どのユーザでも，いずれかの方法を利用することで，判定成功率を 100%にすることができる．

表 8: 全体の実験結果

ユーザ	user1	user8,9	user10
実験 2-1 の判定結果 (なりすましユーザの判定)	100%	11%	-
実験 2-2 の判定結果 (正当なユーザの判定)	-	-	62.4%
実験 3 の判定結果 (正当なユーザの判定)	-	-	100%
実験 4 の判定結果 (なりすましユーザの判定)	0%	100%	-

4 まとめ

本稿では，なりすましを行いシステムに不正侵入するユーザを検出するシステム (IDS-M) を提案した．IDS-M では，正当なユーザの特徴からなる監査データと，ログイン中のユーザのデータを比較し，差異を調べることでユーザの判定を行う．

本稿では，その差異を求める 2 つの方法を考案し評価実験を行った．1 つ目は相関係数を利用する方法であり，2 つ目はキーの入力時間から求められるタイピング間隔を利用する方法である．これらの方法について実際のネットワーク上で評価実験を行った結果，2 つの方法ともユーザ判定に利用できることが示された．

しかし，実際のユーザ判定では本稿で述べた 2 つの方法も含め，複数の方法により総合的に判断して最終的な判定を行う．この際にどの方法に対してどれだけの重みを付けるかなども今後の課題である．

また，本稿で行った実験についても今後も多くの評価実験を行い，閾値 Th の決定と，データ抽出の精密化を行うことが今後の課題である．

参考文献

[1] V. Paxson, "Bro:A System for Detecting Network Intruders in Real-Time," Seventh USENIX Security Symposium, San Antonio, Texas, 1998.

[2] W. Lee, S. J. Stolfo, "Data Mining Approaches for Intrusion Detection," Seventh USENIX Security Symposium, San Antonio, Texas, 1998.