December 1997

Using Time Series Methods in Profiling Computer User Activity for Intrusion Detection: Case Study Results

Peter Best
Queensland University of Technology

George Mohay
Queensland University of Technology

Alison Anderson
Queensland University of Technology

Using Time Series Methods in Profiling Computer User Activity for Intrusion Detection: Case Study Results

Professor Peter Best
School of Accountancy
Queensland University of Technology
GPO Box 2434, Brisbane, 4001, Queensland, Australia
Phone: 617 3864-2739
Fax: 617 3864-1812
Email: p.best@qut.edu.au

Professor George Mohay
School of Computer Science
Queensland University of Technology
GPO Box 2434, Brisbane, 4001, Queensland, Australia
Phone: 617 3864-1964
Fax: 617 3864-1801
Email: g.mohay@qut.edu.au

Ms Alison Anderson
School of Information Systems
Queensland University of Technology
GPO Box 2434, Brisbane, 4001, Queensland, Australia
Phone: 617 3864-2465
Fax: 617 3864-1969
Email: a.anderson@qut.edu.au

Keywords: Computer usage forecasting User profiles Anomaly detection Intrusion detection

Executive Summary
Computer intrusions, such as hacking and viruses, have attracted considerable attention in the research literature and the media. Modern commercial operating systems typically use access control mechanisms to evaluate the legitimacy of requested user activity, rejecting actions that are not authorised. These systems generate voluminous logs of user activity or "audit trails" which record both successful and rejected requested actions. Manual analysis of such records aimed at detecting computer intrusions is usually impractical. This project aims to develop a knowledge base system to facilitate such analysis - a machine-independent audit trail analyser (MIATA).

Various methods have been used by intruders to obtain unauthorised access to on-line systems, including: attempted break-ins, or **password guessing**, where the intruder tries repeatedly to guess a target user's password; **masquerading**, where the intruder logs in to the system as a target user, using his or her user identification (user-id) and password; and **browsing**, or attempts by authorised users to obtain private information (such as user-ids) to assist in the above methods, or to perform unauthorised functions, such as accessing sensitive data files, changing user privileges, printing/displaying large numbers of files and resource hogging.

Four main safeguards against computer intrusions are: authentication, access control, cryptography and audit trail analysis. Authentication, access control and cryptography are aimed at preventing intrusions to computer systems. Analysis of operating system audit trails can provide a powerful tool for detecting unauthorised activity and anomalous user activity. The above intrusion methods may leave evidence recorded in these audit trails of events such as repeated failed logins, logins at unusual times, changes in patterns of terminal, file and command usage by individual users and

across the whole system, repeated unsuccessful attempts to access files and execute commands, particularly those of a sensitive nature.

How would one detect password guessing? Such a break-in attempt could be detected by identifying all users that had any failed logins in a given day, or (since users frequently miskey their passwords) an **anomalous** number of failed logins. In addition, such attempts may be perpetrated using a terminal different from that accessed normally by the target. Selecting users with **anomalous failed logins** and **anomalous terminal usage** could identify quickly users that may have been targeted for password guessing. Similar *profiles* may be developed for other common intrusion methods. A knowledge base system for audit trail analysis can maintain individual user profiles or forecasts of user behaviour that allow the detection of such anomalous behaviour.

A fundamental proposition underlying the development of MIATA is that it is feasible to forecast computer user activity. This requires that suitable forecasting methods can be selected given constraints including timeliness, volumes and storage, which satisfy requirements concerning accuracy and 'good fit'. This paper documents the results of a case study involving experiments with actual user activity data which assesses the effectiveness of time series (smoothing) methods for this purpose.

Abstract
Earlier research on intrusion detection in computer systems has utilised primarily pattern recognition and classification analysis techniques for distinguishing normal and anomalous computer user activity. This project involves the application of forecasting methods to establish computer user profiles which permit the detection of anomalous activities for particular users. This paper documents the results of a case study which assesses the effectiveness of smoothing methods for this purpose.

Introduction
This experiment was associated with a major research project concerning testing the feasibility of implementing a machine-independent audit trail analyser (MIATA). The MIATA project is concerned with **intrusion detection** in multi-user computer systems. There have been a considerable number of intrusion detection projects documented in the literature, such as the Standard Audit File of Garner & Pinnis (1981) and (1984), SRI's Intrusion Detection Expert System (IDES) (Denning 1987; Denning et. al. 1987; and Lunt et. al. 1988), Haystack Laboratories' HAYSTACK System (Smaha 1988), Los Alamos National Laboratory's Wisdom & Sense (W&S) (Liepins & Vaccaro 1989), AT&T's COMPUTERWATCH (Dowell & Ramstedt 1990) and STAT (Ilgun, Kemmerer & Porras 1995). Specialised network intrusion systems are also being developed, such as Planning Research Corporation's Information Security Officer's Assistant (ISOA) (Winkler & Landry 1992), the Internetwork Security Monitor (ISM) (Heberlein, Mukherjee & Levitt 1992) and SRI's NIDES (Jagannathan et. al. 1993).

The MIATA project aims to analyse operating system audit trails on a daily basis to detect activity by computer users which can be categorised as anomalous. Such anomalies may indicate changes in users' organisational responsibilities, but may also indicate attempted break-ins (such as through password guessing) or intruders' masquerading as legitimate users. A necessary component of the MIATA project is the development of user profiles, representing each user's normal expected pattern of activity. Attributes must be forecast for each user on a daily basis such as terminal usage, logins, failed logins, file access and command execution. Forecasting methods are proposed as a novel approach to determine expected frequencies of activity for each user. Methods must be selected which promise reasonable accuracy, minimal storage requirements and minimal computation complexity given the thousands of forecasts which must be generated each day.

This paper reports the results of experiments examining the performance of simple smoothing methods for forecasting the activity of computer users. These results support the application of forecasting methods in intrusion detection systems.

Prior Literature
Four general statistical approaches which may be appropriate for anomaly detection are examined in Javitz et. al. (1993) - pattern recognition, classification analysis, markovian transition analysis and Bayesian decision analysis.

Pattern recognition is a family of techniques which aims to search for structure in data. Cluster analysis is one such technique that has been popular with researchers in the social sciences (especially psychology). It is suggested that **cluster analysis** may not be a feasible approach for distinguishing normal and anomalous actions by users (events). Such an approach would require a substantial research effort to identify relevant variables, storage of large volumes of data (such as the last 10 000 events for each user), and lengthy processing times.

Classification analysis techniques may be an efficient and powerful approach for classifying events as normal or anomalous. However, they require access to large and exhaustive sets of historical examples of normal and anomalous behaviour in order to train the classifier. Such data is not as yet easily accessible. In any case, specific behaviour may be anomalous for one user, but may be quite normal for others.

Markovian transition analysis is concerned typically with the identification of different states and the probabilities of transitions from one state to another. This approach could be used in anomaly detection, whereby an event would represent the transition between two states. The normality of a transition (event) could be assessed by its probability of occurrence. The implementation of Markovian transition analysis would prove expensive, in defining the states of interest for each user, in deriving the large number of transition probabilities for each user, and in writing the software to detect the attainment of a given state.

Bayesian decision analysis may be applicable to rule-based systems that evaluate the probability of an intrusion given the occurrence of a particular sequence of events. Such a system requires incorporation of extensive data on methods of intrusion, target system vulnerabilities and the conditional probabilities that specific observed events will be associated with those methods. This data is not available at present. In addition, Bayesian decision analysis is not likely to be applicable in detecting anomalous user activity. Extensive initial and on-going system training would be required to develop and maintain user profiles incorporating conditional probabilities that specific actions, both successful and failed, will be associated with particular users. At present, the application of Bayesian decision analysis to intrusion and anomaly detection is not well-developed.

Of all the prior research projects involved with audit trail analysis, only the **NIDES** project has published technical details of its statistical methods (see Javitz et. al. 1993; Anderson et. al. 1993; and Javitz & Valdes 1994). In order to detect anomalous user behaviour, NIDES compares a user's short-term behaviour with his/her long-term behaviour. The security officer can specify the *half-life*. For example, if the half-life is 100 audit trail records, the 200th record has only one-quarter the influence of the most recent record, the 400th record has only one-sixteenth of the influence, and so on. A user's activity is characterised as anomalous where short-term behaviour differs from long-term behaviour or where long-term behaviour is absent from short-term behaviour. Accordingly, NIDES uses a form of pattern recognition as its statistical approach. Certain aspects of user behaviour are represented as measures, such as logins, and file accesses. A user's profile consists of a set of values for such measures associated with his/her short-term and long-term behaviour.

Four categories of measures are employed by NIDES: activity intensity, audit record distribution, categorical measures and ordinal measures. Measures of activity intensity are concerned with whether the volume of a specific activity is normal. Audit record distribution measures are concerned with whether types of actions are normal. Categorical and ordinal measures are concerned with whether, within a type of activity, the behaviour over the recent past that affects that action is normal.

NIDES uses a vector called Q that quantifies each measure and this quantification is recorded into a frequency distribution. A historical distribution for Q is built by observing the values of Q over long periods and by selecting appropriate intervals for categorising Q values. Larger values of Q are associated with more anomalous behaviour. NIDES uses an additional vector *S*, that is a transformation of Q, such that *S* is small when Q is small and large when Q is large. This process is a simple mapping of the percentiles of the Q distribution on to the percentiles of a half-normal distribution.

NIDES also uses the T^2 statistic, that is, a summary judgment of the abnormality of many measures (sum of the squares of the individual measures in *S)*. An individual T^2 value is generated for each audit trail record. Larger values indicate more anomalous behaviour. An historical distribution is kept for T^2 values to determine a threshold for anomaly decisions. NIDES requires a training period in order to establish these distributions.

Javitz et. al. (1993) also provides a brief description of the statistical methods used by the WISDOM & SENSE (W&S) and the HAYSTACK projects, and comparisons with the NIDES statistical approach. **W&S** employs classification-tree analysis techniques for anomaly detection. It is the only statistical intrusion detection system that monitors sequences of events. W&S builds its own sequences as a large tree structure where the branches are the sequences and prunes the tree down to contain only the most frequently-used sequences. This tree structure is built for all (not individual) users and incorporates data from only the most recent few days. User activity is compared with the tree to detect anomalous behaviour. When a user has executed a sequence of *n* events, the first *n-1* of which are in the tree, the *nth* event is evaluated to determine whether it is also in the tree. If it is not in the tree, this new event is considered anomalous. In comparison, NIDES bases its anomaly detection decisions on profiles for individual users that incorporate longer-term behaviour.

Little documentation for the **HAYSTACK** statistical approach has been published. Individual measures called *features* are monitored for each user session and compared with historical values to detect abnormality. In comparison, NIDES assesses the normality of each user action on a real-time basis. For each such feature, HAYSTACK determines a range of normal values. HAYSTACK combines the statistical and rule-base components (which are separate in NIDES) in one system. Six generic types of computer abuse are incorporated - break-in, masquerade, penetration of security control system, leakage, denial of service and malicious use. A set of weights (*0-9*) is determined for each such abuse, indicating the extent to which each measure is related to that type of abuse. Each session feature with a value outside the predefined range of normal values causes a weight for that feature to be added to the session's score for that abuse. HAYSTACK calculates the session's abuse score (*suspicion quotient*) and notifies the security officer if the score is too large. No information is available on the measures and weights used by HAYSTACK.

The statistical approaches used by prior researchers have varied widely. No conclusive results are as yet available to justify the selection of one approach over another. Accordingly, the MIATA project uses **forecasting** methods as a novel approach for anomaly detection.

Forecasting Methods For Anomaly Detection

The fundamental proposition for the success of the MIATA project is that it is feasible to forecast user behaviour and to employ such forecasts for categorising user behaviour as normal or anomalous.

Makridakis & Wheelwright (1979) provides a simple framework (see Table 1) for classifying forecasting methods. Two dimensions are utilised for describing forecasting situations - the type of pattern experienced and the type of information available. The type of pattern experienced may be one where history is expected to repeat itself, that is where the historical pattern is expected to continue into the future, or one where the pattern depends on external factors as well as historical observations. In certain forecasting situations, quantitative historical data may be available, such as sales quantities. In other situations, only qualitative data may be available, such as executive opinion.

The upper left-hand quadrant of Table 1 depicts situations which suit time series methods, such as predicting the continuation of sales trends. The lower left-hand quadrant depicts situations where explanatory (or causal) methods are appropriate, such as predicting how prices and advertising impact on sales. The upper right-hand quadrant presents exploratory forecasting methods which are suited to situations where historical patterns of qualitative data are expected to continue, such as predicting motor vehicle speeds in the 21st century. The lower right-hand quadrant represents situations where normative methods are appropriate, that is where the impact of management decision-making is considered in predicting future outcomes.

Table 1 Classification of Forecasting Techniques

Type of Pattern	Quantitative Information	Qualitative Information
History repeats itself	Time series methods Exponential smoothing Decomposition/census II Filters Autoregressive/moving average Leading indicators Various forms of trend extrapolations	Exploratory methods Anticipatory surveys Catastrophe theory Delphi Historical analogies Life-cycle analysis Morphological research Jury of executive opinion Sales force composite
External factors determine events	Explanatory methods Regression Econometric models Multivariate ARMA Input/output	Normative methods Cross-impact matrices Relevance trees Delphi System dynamics Market research

Audit trail analysis is a situation where quantitative information is available. The MIATA project incorporates components which record the frequency of user actions and maintain user profiles which constitute forecasts of the frequency of user actions on a daily basis. Accordingly, the quantitative forecasting methods appear most appropriate for implementation in MIATA.

It is expected that user behaviour will prove to be relatively stable. Users who have been assigned specific job descriptions or roles in commercial organisations are likely to have relatively stable patterns of system usage. They are likely to use primarily the same terminals, and access primarily the same files and programs each day in order to perform their assigned functions. A significant portion of a user's system usage may in fact involve scheduled batch jobs. It is acknowledged that there may be minor random deviations from this stable pattern and that there may be routine deviations on a periodic basis, such as at the end of the working week, at the end of the month, and at the end of the financial year.

In addition, the privileges assigned to specific users should also reflect their roles. Linked to the principle of *separation of duties* (NCSC 1985), the security principle of *least privilege* requires that a user should have access to the fewest objects (files and programs) needed to perform the functions associated with his/her role. Furthermore, access to information should be limited by the *need-to-know* rule: access to sensitive data should be allowed only to users needing that access to perform their role (Pfleeger 1989). An organisation's security system is likely to be most effective if these concepts are incorporated. If implemented, these concepts also have the effect of stabilising the activity of individual system users. Given these arguments, it is proposed that basing forecasting

method selection on the assumption that the pattern of historical behaviour will repeat itself is quite supportable.

Explanatory or *causal* methods such as regression assume that the factor to be forecast exhibits a cause-effect relationship with one or more independent variables. Such methods determine the form of this relationship and use it to forecast future values of the dependent variable given observations of the independent variable(s). In contrast, *time series* methods assume that future values of a factor can be predicted merely from historical values of the factor and/or past forecasting errors. Such methods determine the pattern in these past values and extrapolate that pattern into the future.

In the current context, time series methods offer a practical advantage over explanatory or causal methods. In order to generate a forecast, the system requires access only to historical values and/or errors. There is no need to identify the factors (independent variables) influencing user behaviour nor to determine the nature of that influence. For the MIATA project, the selection of suitable time series forecasting methods should consider the nature of the forecasting situation and the characteristics of the alternative methods. The MIATA project involves the generation of immediate time horizon forecasts, that is, daily forecasts of the frequency of user activity. Such a time horizon reduces severely the impact of data patterns with trend, seasonal or cyclical characteristics. Because users are performing stable external organisational roles, it is expected also that case studies of actual user behaviour will demonstrate the relative stationarity of the data series. Superior accuracy may be achievable using more sophisticated methods such as decomposition and ARMA. However, the forecasting situation faced is one where tens of thousands of items must be forecast on a daily basis, where data storage requirements must be minimised, where development and running costs must be minimised and where complexity must be kept to a reasonable level.

Forecasting methods that require retention and analysis of historical values for each data series would impose substantial costs on the MIATA system. Decomposition and ARMA methods have such data storage requirements and generate forecasts most efficiently using specialised software packages. In order for MIATA to incorporate such methods, the relevant data series would have to be transferred using an intermediate file to the relevant external package and the forecast generated would have to be transferred back to MIATA. Such methods may prove impractical where they involve human involvement in recognising data patterns, model selection and optimisation of parameters.

Given the above arguments, only simple smoothing methods were recommended for potential incorporation in MIATA. These methods are incremental in nature, in that they generate forecasts from prior forecasts and errors. Such methods are suitable for immediate time horizon forecasting for relatively stationary data series, and impose minimal costs in terms of development, data storage and running costs.

The Experiment

The objective of this experiment was to compare the performance of simple smoothing time series methods in forecasting actual user behaviour. This section describes a case study aimed at comparing the effectiveness of the selected forecasting methods.

Forecasts were generated using each of six time series methods for each data series. The accuracy of each of five smoothing methods - adaptive-response-rate single exponential smoothing (ARRSES), Brown's one-parameter linear exponential smoothing (BROWN), simple average (MEAN), single moving average of order 5 (MOVAV), and single exponential smoothing (SES) - is evaluated relative to each other and to the naive method (NAIVE). For each data series, the method which minimises forecasting error was identified. The error measures used in comparing methods were the Mean Squared Error (MSE) and the Mean Absolute Percentage Error (MAPE). The "goodness of fit" of the more accurate method was then assessed by examining the autocorrelation coefficients (ACFs) of the errors.

The case study employed data from the Queensland Railways (QRAIL) IBM installation running the MVS/ESA operating system environment. The primary processing applications at QRAIL are payroll/personnel, wagon tracing, freight accounting, seat/berth reservations, crew rostering, locomotive maintenance and supply. These applications are maintained using a COBOL/SQL/CICS programming environment and IBM's IMS/DB2 database management systems. There are over 5500 interactive users of the QRAIL system employing over 1700 terminals.

A set of 20 users was selected for monitoring with the assistance of Information Systems Audit staff. Users were selected with high volumes of activity and a wide range of action-on-object events. Seven weeks' data was collected for the 20 users. After removing weekends and public holidays, user activity data for 32 weekdays was available for conducting the forecasting experiment.

It was decided early to restrict these experiments to week-day data. Relatively few users were active weekend users. Preliminary experiments indicated that better results would be obtained by operating two versions of MIATA, one to monitor week-day user activity and another to monitor weekend user activity. The absence of activity on weekends appeared to bias forecasts downwards and generate primarily positive forecasting errors on week-days and negative errors on weekends.

Figure 1 summarises the composition of the 220 data series chosen for testing the effectiveness of the selected forecasting methods. These data series focused on the frequency of individual user's actions on objects. Successful logins, late logins and failed logins were collected for each of the 20 users. Each user's failed actions data series was also collected. Another 44 data series dealt with individual users' terminal usage - successful and failed logins. Sixty-six (66) terminals were used by these 20 users over the 32-day period, thereby giving 132 (66 successful logins and 66 failed logins series) potential data series for analysis. User-terminal relationships were selected with at least 15 successful logins over the 32 days (22 of the 66 terminal usage relationships). Both successful logins and failed logins data series were analysed for these relationships.

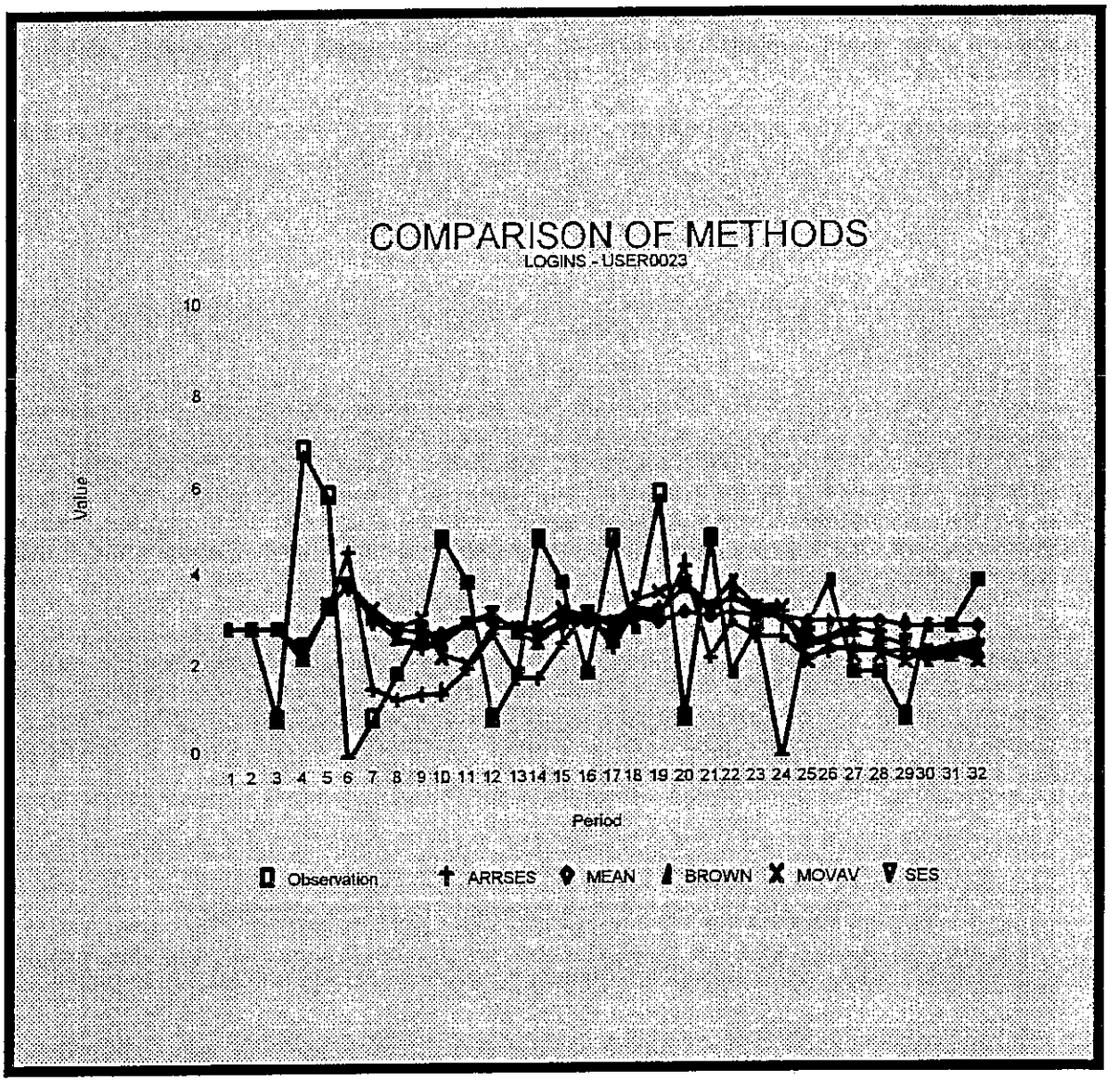
The 20 users attempted to access (read, write, failed read or failed write) 321 files over the 32 days, giving 1284 potential data series. File usage data series were selected with at least 20 actions over the 32 days resulting in 89 data series for analysis. Only 19 different commands executed by these 20 users over the 32-day period were monitored by RACF and recorded in the operating system audit trail. Accordingly, relatively few command usage data series were available for analysis. Of the 38 potential data series (executes and failed executes), only those with at least 5 actions over the 32 days were selected for analysis (7 data series).

Figure 1 Summary of Case Study 1 Data Series

Nature of Series	*No. of Series*
For individual users:	
Login Activity	
Successful logins	20
Failed logins	20
Late logins	20
Failed Activity	
Failed Actions	20
Terminal Usage	
Successful terminal logins	22
Failed terminal logins	22
File Usage	
Successful file reads	82
Failed file reads	5
Successful file writes	2
Command Usage	
Successful command execs.	7
TOTAL DATA SERIES	220

The forecasting experiments were conducted using the LOTUS 1-2-3 spreadsheet - FORECAST. The FORECAST spreadsheet performs a simulation of the six forecasting methods for any given data series. The following parameters were used: ARRSES - a_1=.2 and β=.2; BROWN - a=.1; SES - a=.2. FORECAST simulates the generation of the daily forecasts for each of the six methods, measuring daily forecast errors under each method, and printing a summary report. This report lists the data series with the resulting forecast errors and summarises the MSE and MAPE for each method. This output allows the identification of the method which performs best for each data series, on the basis of each measure of forecasting accuracy, and provides the error data to permit an analysis of the "goodness of fit" for the superior method. FORECAST also provides the facility to display or print a graphical representation of the performance of the methods for a given data series. Figure 2 provides sample output.

Figure 2 Graphical Comparison of Forecasting Methods



Each data series was subdivided into an initialisation set of 5 observations and a test set used in comparing forecasting accuracy on the basis of MSE and MAPE. Figure 3 shows the relative performance of these methods in terms of forecasting accuracy, that is, in minimising MSE and MAPE. Figure 3 indicates the number of data series for which each of the six methods performed best in minimising MSE and MAPE, respectively. The mean method minimised MSE and MAPE for the greatest proportion of the data series. This result suggests that these data series exhibit a high degree of stationarity and provides support for the focus on smoothing methods.

As is evident in Figure 3, the five smoothing methods outperformed the naive method in terms of minimising MSE and MAPE. The measures of MSE and MAPE for the smoothing methods did not vary greatly for individual data series, causing a significant number of ties when the performance of these methods was ranked. Seventy-nine percent (173) of the data series had a MSE range (maximum MSE less minimum MSE among the smoothing methods) for the 32 days of less than 5, causing little variation over the five methods in *standard error*, that is, less than 2.2, and little variation in associated confidence intervals for a data series among these methods. (MIATA uses each data series' standard error to derive confidence intervals).

The randomness of the mean method's forecasting errors was examined in order to test its "goodness of fit". For 95 percent confidence, all ACFs should lie within the range (where $n=31$):

$$(4.2) \quad I_{95\%} = 0 \pm \frac{1.96}{\sqrt{n}}$$

$$= 0 \pm .352$$

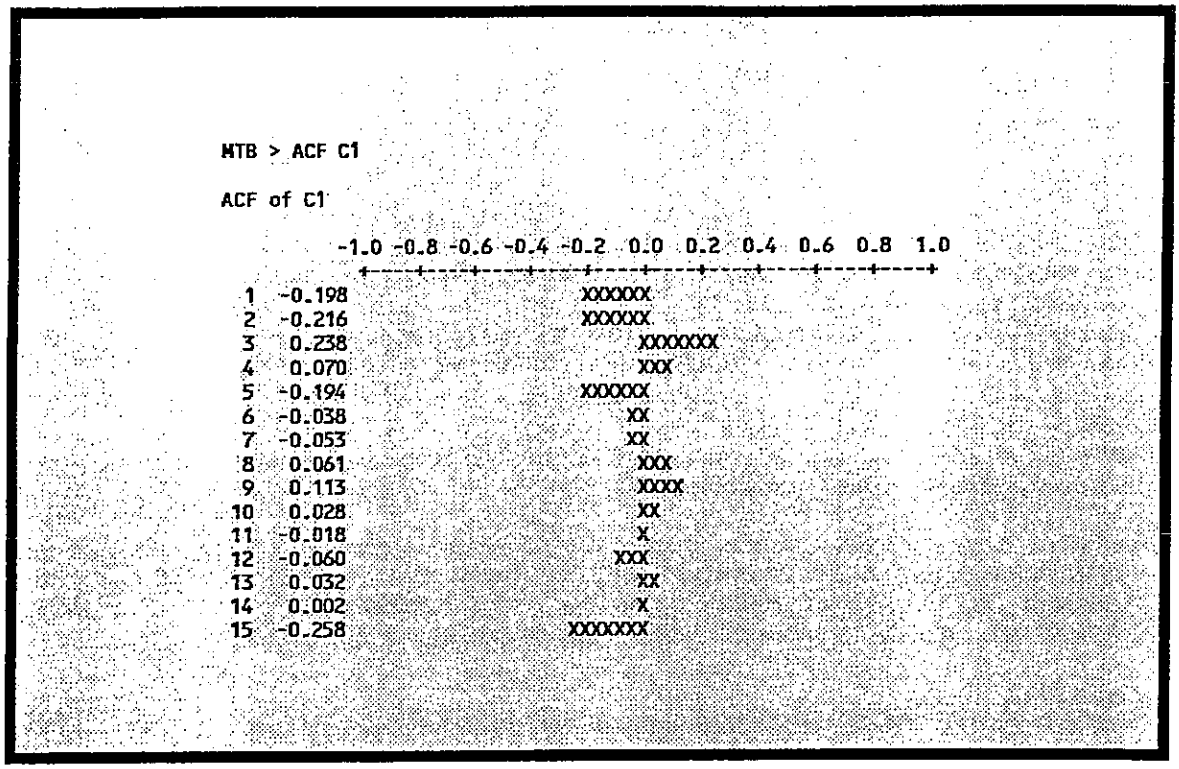
Seventy-six percent (167) of the 220 ACF plots for the forecast errors produced by the mean method satisfied this criterion. The mean method provided a "good fit" for 76 percent of the data series. Fifty-three of the plots showed ACFs lying marginally outside the required interval. The histogram plots for the forecast errors were also examined for consistency with a normal distribution with a zero mean and absence of consistent observable skewness. Eighty-five percent (188) of the histograms appeared normal. Some minor skewness was apparent in 32 of the plots. Figures 4 and 5 provide sample output from these plots.

Similar results were obtained with two further case studies. These studies involved two different organisations (a government service bureau and a tertiary institution) with different computing environments (UNISYS OS/1100 and VAX VMS). While not conclusive, these results provide support for the application of smoothing time series methods to forecasting computer user behaviour and suggest that such forecasts (with their associated confidence intervals) can be employed to successfully categorise user behaviour as normal or anomalous.

Figure 3 Case Study - Comparing Forecasting Methods

Method	*No. of Series with Min. MSE*	*No. of Series with Min. MAPE*
ARRSES	139	77
BROWN	123	90
MEAN	165	123
MOVAV	119	91
NAIVE	78	69
SES	153	76
TOTAL	220	220

Figure 4 Sample Output from ACFs Plot

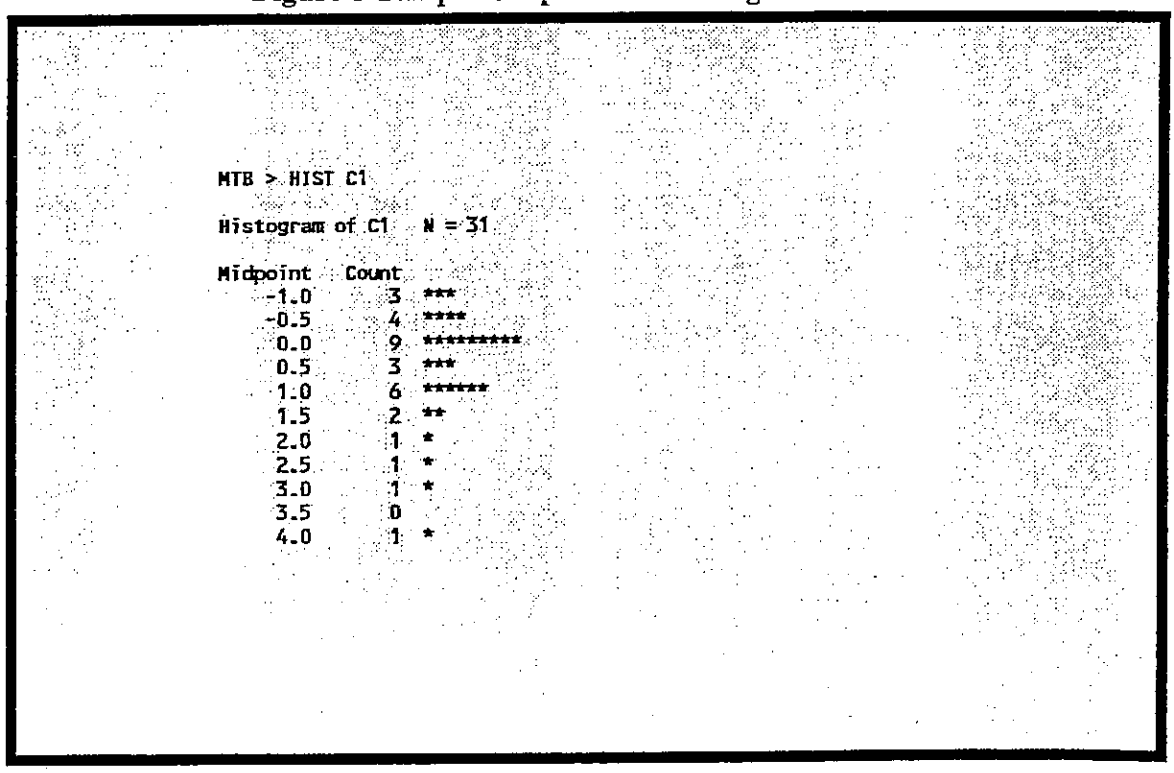


```
MTB > ACF C1

ACF of C1

          -1.0 -0.8 -0.6 -0.4 -0.2  0.0  0.2  0.4  0.6  0.8  1.0
          +----+----+----+----+----+----+----+----+----+----+----+
 1  -0.198                    XXXXXX
 2  -0.216                    XXXXXX
 3   0.238                         XXXXXXX
 4   0.070                         XXX
 5  -0.194                    XXXXXX
 6  -0.038                       XX
 7  -0.053                       XX
 8   0.061                        XXX
 9   0.113                        XXXX
10   0.028                        XX
11  -0.018                        X
12  -0.060                      XXX
13   0.032                        XX
14   0.002                        X
15  -0.258                    XXXXXXX
```

Figure 5 Sample Output from Histogram Plot



```
MTB > HIST C1

Histogram of C1   N = 31

Midpoint   Count
    -1.0       3  ***
    -0.5       4  ****
     0.0       9  *********
     0.5       3  ***
     1.0       6  ******
     1.5       2  **
     2.0       1  *
     2.5       1  *
     3.0       1  *
     3.5       0
     4.0       1  *
```

Sparse Data Series
A significant number of data series were excluded from forecasting experiments in the case studies. These data series could be categorised as *sparse*, in that they involved few (such as less than 5) observations over the 32-day period. These data series had zero observations (that is, no successful reads of a particular file, or no failed logins on a particular terminal) on at least 27 of the 32 week-days. Some comments are provided below concerning the significance of these exclusions and the performance of the naive and smoothing forecasting methods on these data series.

Sparse data series, such as failed logins or failed writes, involve a large number of zero observations and typically a few non-zero observations of low magnitude. Accordingly, forecasts and standard errors (and associated confidence intervals) tend to be close to zero. Minimal variation in forecasting accuracy is observable among the six methods. The smoothing methods however appear to consistently outperform the naive method. Since the majority of forecast errors (at least 27 of the 32) are close to zero, the requirement for the errors to be normally distributed around a mean of zero will be satisfied by all methods. In cases where the nonzero observations are relatively close to zero, the ACFs will satisfy the requirements for a good fit. The ACFs however may be distorted where the non-zero observations are large in magnitude.

It may be concluded therefore that the smoothing method selected for incorporation in the MIATA system on the basis of the above case studies will also be suitable for forecasting and detecting anomalous levels of activity for these sparse data series. Non-zero observations on a particular day are likely to exceed their confidence intervals and be categorised as anomalies.

Limitations
A number of potential limitations can be identified where forecasting methods are used in intrusion detection systems.

The MIATA reporting facilities identify users who meet the activity profiles likely to be associated with intrusion methods. For example, password guessing may be indicated by either anomalous failed logins, or a combination of failed logins and anomalous terminal usage.

The effectiveness of the MIATA system in detecting such an intrusion depends on its ability to identify **anomalies**. MIATA's forecasting and anomaly detection component derives confidence intervals which are compared with actual observations. These confidence intervals are determined with reference to the forecast (F_t), the standard error (SE_t) and a specified number of standard errors (d).

It is proposed that Type II errors assume greater importance than Type I errors in the design of the MIATA system. Failing to detect anomalous activity is seen to be a primary concern of the MIATA system. Accordingly, it is argued that a conservative value for d (such as 1) may be more effective. However, it is recognised that such an approach requires the provision of reporting facilities permitting effective use of the security officer's time in investigating the larger number of anomalies that would be reported.

The effectiveness of the MIATA system in detecting actual intrusion activity also depends on the characteristics of the user targeted by the intruder. Where the target's data series are relatively stable, the corresponding confidence intervals will be narrower than if those data series are more variable. Additional actions by the intruder who is masquerading as the authorised user may be reported or not reported as anomalous depending on the width of those confidence intervals. However, all activity attempted by the intruder which is new for the authorised user will be categorised and reported as anomalous.

MIATA's security officer reporting component focuses attention on users meeting the activity profiles associated with the intrusion methods and on users with larger volumes of anomalous actions. However, a more sophisticated intruder may be aware or cautious of potential monitoring activities and accordingly concentrate on low volume activity. Given the above discussion, such an intruder may target relatively busy users with more variable data series and avoid detection.

Genuine anomalies resulting from random events, special projects or end-of-period tasks may distort significantly the confidence intervals for particular data series. These confidence intervals may remain high for some time as a result and conceal potential intrusions. Ideally, the security officer could be capable of investigating each anomaly and providing input to the system to specify whether it should be incorporated in the data series as an observation or omitted to prevent distortion of confidence intervals. Such an approach is not considered feasible at this stage.

Conclusion

The results of the case study documented in this paper support the propositions that simple smoothing time series methods may be effective for developing computer user profiles, and that computer user activity can be categorised successfully as normal or anomalous using forecasting methods. A wide range of user activity was monitored over 32 week-days and used in forecasting experiments to assess the effectiveness of smoothing methods.

In the case study, the naive and five smoothing methods were ranked on the basis of forecasting accuracy, in terms of minimising MSE and MAPE. The five smoothing methods outperformed the naive method. The mean method appeared superior consistently, and analyses of the resulting forecasting errors indicated that this method provided a "good fit" for the majority of the data series. Accordingly, the mean method is recommended for initial incorporation in the MIATA system.

It is acknowledged that these results are based on samples of user activity over relatively short periods of time and are not necessarily generalisable to populations, such as all users or all computer installations. However, these encouraging results provide support for the proposition that computer user behaviour is forecastable. It is suggested that further case study research should be conducted to accumulate further compelling evidence supporting this proposition.

Bibliography

Anderson, D., Lunt, T.F., Javitz, H., Tamaru, A. & Valdes, A. Safeguard Final Report: Detecting Unusual Program Behavior Using the NIDES Statistical Component. Final Report, SRI International, Menlo Park, CA., 1993.

Denning, D.E. "An Intrusion Detection Model". IEEE Transactions on Software Engineering, Vol. SE-13, No.2, February, 1987, pp. 222-232.

Denning, D.E., Edwards, D.E., Jagannathan, R., Lunt, T.F. & Neumann, P.D. A Prototype IDES: A Real-Time Intrusion-Detection Expert System. SRI International, Menlo Park, CA., 1987.

Dowell, C. & Ramstedt, P. "The COMPUTERWATCH Data Reduction Tool". Proc. 13th National Computer Security Conference, Baltimore, MD, October, 1990, pp. 99-108.

Garner, B.J. & Pinnis, J. "A Standard Audit File and its Implications for the EDP Auditor". Proc. 1981 EDP Auditors Association 3rd National Conference, EDP Auditors Association, Melbourne, 1981.

Garner, B.J. & Pinnis, J. "Modelling as an Audit Technique". The Australian Computer Journal, Vol. 16, No. 2, 1984, pp. 48-53.

Heberlein, L.T., Mukherjee, B. & Levitt, K.N. "Internetwork Security Monitor: An Intrusion-Detection System for Large-Scale Networks". Proc. 15th National Computer Security Conference, Baltimore, MD, October, 1992, pp. 262-271.

Ilgun, K., Kemmerer, R.A. & Porras, P.A. "State Transition Analysis: A Rule-Based Intrusion Detection Approach". IEEE Transactions on Software Engineering [ISO], Vol. 21, No. 3, 1995, pp 181-199.

Jagannathan, R., Lunt, T., Anderson, D., Dodd, C., Gilham, F., Jalali, C., Javitz, H., Neumann, P., Tamaru, A. & Valdes, A. System Design Document: Next-Generation Intrusion Detection Expert System (NIDES). SRI International, Menlo Park, CA., 1993.

Javitz, H.S. & Valdes, A. The NIDES Statistical Component: Description and Justification. SRI International, Menlo Park, CA., 1994.

Javitz, H.S., Valdes, A., Lunt, T.F., Tamaru, A., Tyson, M. & Lowrance, J. Next Generation Intrusion Detection Expert System (NIDES): 1. Statistical Algorithms Rationale; 2. Rationale for Proposed Resolver. SRI International, Menlo Park, CA., 1993.

Liepins, G.E. & Vaccaro, H.S. "Anomaly Detection: Purpose and Framework". Proc. Nat. Comp. Security Convention, 1989, pp. 495-504.

Lunt, T.F., Jagannathan, R., Lee, R., Listgarten, S., Edwards, D.L., Neumann, P.G., Javitz, H.S. & Valdes, A. IDES: The Enhanced Prototype - A Real-Time Intrusion-Detection Expert System. SRI-CSL-88-12, SRI International, Menlo Park, CA., 1988.

Makridakis, S., Wheelwright, S.C. & McGee, V.E. Forecasting: Methods and Applications. 2nd edn, Wiley, New York, 1983.

National Computer Security Center DoD Trusted Computer System Evaluation Criteria. DoD 5200.28-STD, DoD, 1985.

Pfleeger, C.P. Security in Computing. Prentice-Hall, Englewood Cliffs, NJ., 1989.

Smaha, S.E. "Haystack: An Intrusion Detection System". Fourth Aerospace Computer Security Applications Conference, 1988, pp. 37-44.

Winkler, J.R. & Landry, J.C. "Intrusion and Anomaly Detection: ISOA Update". Proc. 15th National Computer Security Conference, Baltimore, MD, October, 1992, pp. 272-281.