

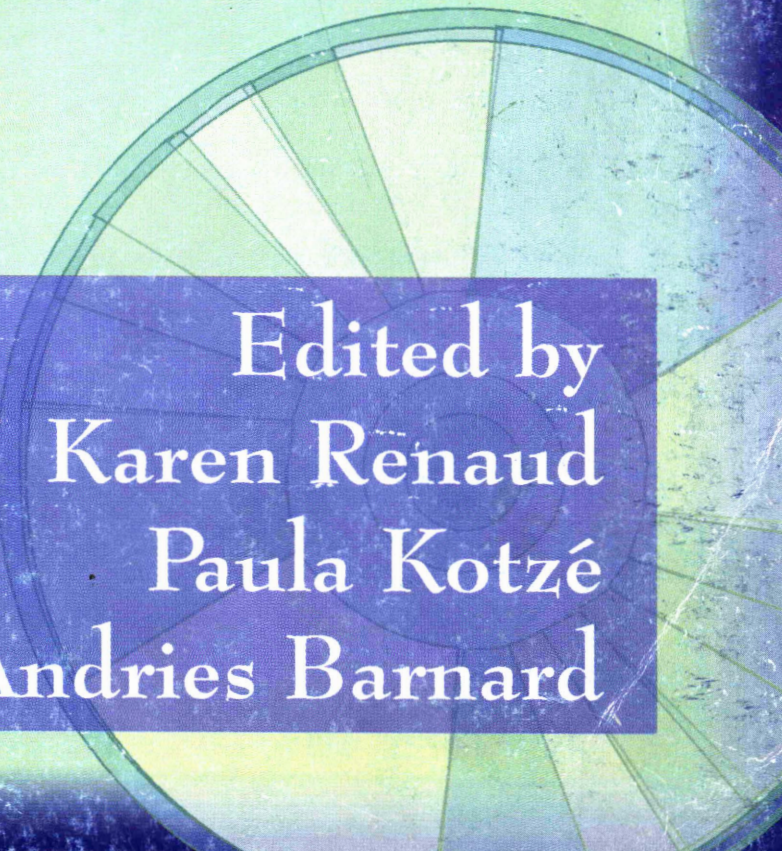
HARDWARE, SOFTWARE AND PEOPLEWARE



UNISA



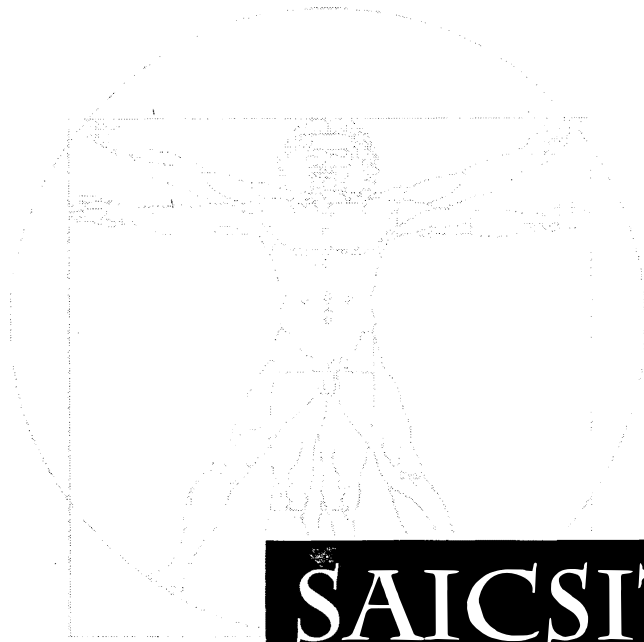
SAICSIT 2001



Edited by
Karen Renaud
Paula Kotzé
Andries Barnard

HARDWARE, SOFTWARE AND PEOPLEWARE

**South African Institute of Computer
Scientists and Information Technologists**
Annual Conference
25 – 28 September 2001
Pretoria, South Africa



SAICSIT 2001



Edited by Karen Renaud, Paula Kotzé & Andries Barnard
University of South Africa, Pretoria

Proceedings of the Annual Conference of the South African Institute of Computer Scientists and Information Technologists

**First Edition, First Impression
ISBN: 1-86888-195-4**

© The South African Institute of Computer Scientists and Information Technologists (SAICSIT)

Abstracting is permitted with credit to the source. Liberties are permitted to photocopying beyond the limits of South African copyright law for private use for research purposes. For other photocopying, reprint or republication permission write to the SAICSIT President, Department of Computer Science and Information Systems, UNISA, P O Box 392, Pretoria, 0003, South Africa.

The Publisher makes no representation, expressed or implied, with regard to the accuracy of the information contained in this book and cannot accept liability for any errors or omissions that may be made. The Publisher is not responsible for the use which might be made of the contents of this book.

Published by Unisa Press
University of South Africa
P O Box 392, Pretoria, 0003

Cover Design by Tersia Parsons

Editors: Karen Renaud, Paula Kotzé & Andries Barnard

Electronic Publication by the Editors

Printed by Unisa Press
2001

Table of Contents

Message from the SAICSIT President	iv
Message from the Chairs	vi
Conference Organisation	vii
Referees	viii

Keynote Speakers

<i>Cyber-economies and the Real World</i>	<i>xi</i>
Alan Dix	
<i>Computer-aided Instruction with Emphasis on Language Learning</i>	<i>xiv</i>
Lut Baten	
<i>Internet and Security Trends.....</i>	<i>xv</i>
Arthur Goldstuck	
<i>The Future of Data Compression in E-technology</i>	<i>xvi</i>
Nigel Horspool	
<i>Strategic Planning for E-Commerce Systems: Towards an Inspirational Focus</i>	<i>xvii</i>
Raymond Hackney	

Research Papers

Human-Computer Interaction / Virtual Reality

<i>The Development of a User Classification Model for a Multi-cultural Society.....</i>	<i>1</i>
M Streicher, J Wesson & A Calitz	
<i>Real-Time Facial Animation for Virtual Characters.....</i>	<i>11</i>
D Burford & E Blake	
<i>The Effects of Avatars on Co-presence in a Collaborative Virtual Environment.....</i>	<i>19</i>
J Casanueva & E Blake	

Education

<i>Structured Mapping of Digital Learning Systems.....</i>	<i>29</i>
E Cloete & L Miller	

Formal Methods

<i>The specification of a multi-level marketing business</i>	<i>35</i>
A van der Poll & P Kotzé	
<i>Finite state computational morphology - the case of the Zulu noun</i>	<i>45</i>
L Pretorius & S Bosch	
<i>Combining context provisions with graph grammar rewriting rules: the three-dimensional case</i>	<i>54</i>
A Barnard & E Ehlers	

Human-Computer Interaction / Web Usability

<i>Web Site Readability and Navigation Techniques: An Empirical Study.....</i>	<i>64</i>
P Licker, R Anderson, C Macintosh & A van Kets	
<i>Jiminy: Helping Users to Remember Their Passwords</i>	<i>73</i>
K Renaud & E Smith	

Information Security

<i>Computer Security: Hacking Tendencies, Criteria and Solutions.....</i>	<i>81</i>
M Botha & R von Solms	
<i>An access control architecture for XML documents in workflow environments</i>	<i>88</i>
R Botha & J Eloff	

Graphics and Ethics

<i>Model-based Segmentation of CT Images</i>	<i>96</i>
O Marte & P Marais	
<i>Towards Teaching Computer Ethics.....</i>	<i>102</i>
C de Ridder, L Pretorius & A Barnard	

Human-Computer Interaction / Mobile Devices

<i>Ubiquitous Computing and Cellular Handset Interfaces – are menus the best way forward?</i>	<i>111</i>
G Marsden & M Jones	
<i>A Comparison of the Interface Effect on the Use of Mobile Devices.....</i>	<i>120</i>
J Franken, A Stander, Z Booley, Z Isaacs & R Rose	
<i>The Effect of Colour, Luminance, Contrast, Icons, Forgiveness and Closure on ATM Interface Efficiency</i>	<i>129</i>
A Stander, P van der Zee, & Y Wang	

Object Orientation

<i>JavaCloak - Considering the Limitations of Proxies for Facilitating Java Runtime Specialisation</i>	<i>139</i>
K Renaud	

Hardware

<i>Hierarchical Level of Detail Optimization for Constant Frame Rate Rendering.....</i>	<i>147</i>
S Nirenstein, E Blake, S Windberg & A Mason	
<i>A Proposal for Dynamic Access Lists for TCP/IP Packet Filtering</i>	<i>156</i>
S Hazelhurst	

Information Systems

<i>The Use of Technology to Support Group Decision-Making in South Africa</i>	<i>165</i>
J Nash, D Gwilt, A Ludwig & K Shaw	
<i>Creating high Performance I.S. Teams</i>	<i>172</i>
D C Smith, M Becker, J Burns-Howell & J Kyriakides	
<i>Issues Affecting the Adoption of Data Mining in South Africa</i>	<i>182</i>
M Hart, E Barker-Goldie, K Davies & A Theron	

Information Systems / Management

<i>Knowledge management: do we do what we preach?</i>	<i>191</i>
M Handzic, C Van Toorn, & P Parkin	
<i>Information Systems Strategic Planning and IS Function Performance: An Empirical Study</i>	<i>197</i>
J Cohen	

Formal Methods

<i>Implication in three-valued logics of partial information</i>	<i>207</i>
A Britz	
<i>Optimal Multi-splitting of Numeric value ranges for Decision Tree Induction</i>	<i>212</i>
P Lutu	

Abstracts of Electronic Papers

<i>Lessons learnt from an action research project running groupwork activities on the Internet: Lecturers' experiences</i>	221
T Thomas & S Brown	
<i>A conceptual model for tracking a learners' progress in an outcomes-based environment</i>	221
R Harmse & T Thomas	
<i>Introductory IT at a Tertiary Level – Is ICDL the Answer?</i>	222
C Dixie & J Wesson	
<i>Formal usability testing – Informing design</i>	222
D van Greunen & J Wesson	
<i>Effectively Exploiting Server Log Information for Large Scale Web Sites</i>	223
B Wong & G Marsden	
<i>Best Practices: An Information Security Development Trend</i>	223
E von Solms & J Eloff	
<i>A Pattern Architecture, Using patterns to define an overall systems architecture</i>	224
J van Zyl & A Walker	
<i>Real-time performance of OPC</i>	224
S Kew, & B Dwolatzky	
<i>The Case for a Multiprocessor on a Die: MoaD</i>	225
P Machanick	
<i>Further Cache and TLB Investigation of the RAMpage Memory Hierarchy</i>	225
P Machanick & Z Patel	
<i>The Influence of Facilitation in a Group Decision Support Systems Environment</i>	226
T Nepal & D Petkov	
<i>Managing the operational implications of Information Systems</i>	226
B Potgieter	
<i>Finding Adjacencies in Non-Overlapping Polygons</i>	226
J Adler, GD Christelis, JA Deneys, GD Konidaris, G Lewis, AG Lipson, RL Phillips, DK Scott-Dawkins, DA Shell, BV Strydom, WM Trakman & LD Van Gool	

Message from the SAICSIT President

The South African Institute of Computer Scientists and Information Technologists (SAICSIT) was formed in 1982 and focuses on research and development in all fields of computing and information technology in South Africa. Now in the 20th year of its existence, SAICSIT has come of age, and through its flagship series of annual conferences provides a showcase of not only the best research from the Southern-African region, but also of international research, attracting contributions from far afield. SAICSIT does, however, not exist or operate in isolation.

More than 50 years have passed since the first electronic computer appeared in our society. In the intervening years technological development has been exponential. Over the last 20 years there has been a vast growth and pervasiveness of computing and information technology throughout the world. This has led into the expansion and consolidation of research into a diversity of new technologies and applications in diverse cultural environments. During this period huge strides have also been made in the development of computing devices. The processing speed of computers has increased thousand-fold and memory capacity from megabytes to gigabytes in the last decade alone. The Southern African region did not miss out on these developments.

It is hardly possible for such quantitative expansion not to bring a change in quality. Initially computers had been developed mainly for purposes such as automation for the improvement of processing, labour-reduction in production and automation control of machinery, with artificial intelligence, which made great strides in the 1980s, seen as the ultimate field to which computers could be applied. As we moved into the 1990s it was recognized that such an automation route was not the only direction in the improvement of computers. The expansion of processing power has enabled image data to be incorporated into computer systems, mainly for the purpose of improving human utilisation. For most computer technologies of the 1990s, including the Internet and virtual reality, automation was not the ultimate purpose. Humans were increasingly actively involved in the information-processing loop. This involvement has gradually increased as we move into the 21st century. Development of computer technology based not on automation, but on interaction, is now fully established.

The method of interaction has significantly changed as well. The expansion of computer ability means that the same function can be performed far more cheaply and on smaller computers than ever before. The advent of portable and mobile computers and pervasive computing devices is ample evidence of this. The need for users to be at the same location as a computer in order to reap the benefits of software installed on that computer is becoming an obsolete notion. Time and space are no longer constraints. One of the most discussed impacts of computing and information technology is *communication* and the easy accessibility of information. This changes the emphasis for research and development – issues such as cultural, political, and economic differences must, for example, be accommodated in ways that researchers have not previously considered. Our goal should be to enable users to benefit from technological advances, hence matching the skills, needs, and expectations of users of available technologies to their immense possibilities.

The conference theme for the SAICSIT 2001 Conference – *Hardware, Software and Peopleware: The Reality in the Real Millennium* – aims to reflect technological developments in all aspects related to computerised systems or computing devices, and especially reflect the fact that each influences the others.

Not only has SAICSIT come of age in the 21st century, but so has the research and development community in Southern Africa. The outstanding quality of papers submitted to SAICSIT 2001, of which only a small selection is published in this collection, illustrates both the exciting and developing nature of the field in our region. I hope that you will enjoy SAICSIT 2001 and that it will provide opportunities to cultivate and grow the seeds of discussion on innovative and new developments in computing and information technology.

Paula Kotzé
SAICSIT President

Message from the Chairs

Running this conference has been rewarding, exciting and exhausting. The response to the call for papers we sent out in March was overwhelming. We received 64 paper submissions for our main conference and twelve for the postgraduate symposium. We had a panel of internationally recognized reviewers, both local and international. The response from the reviewers was impressive – accepting a variety of papers and *mostly* returning the reviews long before the due date. We were struck, once again, by the sheer magnanimity of academia – as busy as we all are, we still manage to contribute fully to a conference such as SAICSIT.

After an exhaustive review process, where each paper was reviewed by at least three reviewers, the program committee accepted 26 full research papers and 14 electronic papers. Five papers were referred to the postgraduate symposium, since they represented work in progress – not yet ready for presentation to a full conference but which nevertheless represented sound and relevant research. The papers published in this volume therefore represent research of an internationally high standard and we are proud to publish it. Full electronic papers will be available on the conference web site (<http://www.cs.unisa.ac.za/saicsit2001/>).

Computer Science and Information Systems academics in South Africa labour under difficult circumstances. *The popularity of IT courses stems from the fact that IT qualifications are in high demand in industry, which leads in turn to a shortage of IT academic staff to teach the courses, even when posts are available. The net result is that fewer people teach more courses to more students. IT departments thus rake in ever-increasing amounts of state subsidy for their universities. These profits, euphemistically labelled “contribution to overhead costs”, are deployed in various ways: cross-subsidization of non-profitable departments; maintenance of general facilities; salaries for administrative personnel, etc. Sweeteners of generous physical resources for the IT departments may be provided. We have yet to hear of a University in South Africa where significant concessions have been made in terms of industry-related remuneration. At best, small subventions are provided. As a result, shortages of quality staff remain acute in most IT departments – especially at senior teaching levels. What is even worse is that academics in these departments have to motivate the value of their conference contributions and other IT outputs to selection committees, often dominated by sceptical academic power-brokers from the more traditional departments whose continued survival is underwritten by IT’s contribution to overhead costs.*¹

The papers published in this volume are conclusive evidence of the indefatigability and pertinacity of Computer Science and Information Systems academics and technologists in South Africa. We are proud to be part of such a prestigious and innovative group of people.

In conclusion, we would like to thank the conference chair, Prof Paula Kotzé, for her support. We also specially thank Prof Derrick Kourie for his substantial contribution. Finally, to all of you, contributors, presenters, reviewers and organisers – a big thank you – without you this conference could not be successful.

Enjoy the Conference!

Karen Renaud & Andries Barnard

¹ This taken almost verbatim from Professor Derrick Kourie’s SACLA 2001 paper titled: “*The Benefits of Bad Teaching*”.

Conference Organisation

General Chair

Paula Kotzé

Programme Chairs

Karen Renaud
Andries Barnard

Organising Committee Chairs

Lucas Venter, Alta van der Merwe

Art and Design

Tersia Parsons

Sponsor Liaison

Paula Kotzé, Chris Bornman

Secretarial & Finances

Christa Prinsloo, Elmarie Havenga

Marketing & Public Relations

Klarissa Engelbrecht, Elmarie van
Solms, Adriaan Pottas, Mac van der
Merwe

Audio Visual

Tobie van Dyk, Andre van der Poll,
Mac van der Merwe

Program Committee

Bob Baber – McMaster University, Canada
Andries Barnard – University of South Africa
Judy Bishop – University of Pretoria
Andy Bytheway – University of the Western Cape
Andre Calitz – University of Port Elizabeth
Elsabe Cloete – University of South Africa
Carina de Villiers – University of Pretoria
Alan Dix – Lancaster University, United Kingdom
Jan Eloff – Rand Afrikaans University
Andries Engelbrecht – University of Pretoria
Chris Johnson – University of Glasgow, United Kingdom
Paul Licker – University of Cape Town
Paula Kotzé – University of South Africa
Derrick Kourie – University of Pretoria
Philip Machanick – University of the Witwatersrand
Gary Marsden – University of Cape Town
Don Petkov – University of Natal in Pietermaritzburg
Karen Renaud – University of South Africa
Ian Sanders – University of the Witwatersrand
Derrick Smith – University of Cape Town
Harold Thimbleby – Middlesex University, United Kingdom
Theda Thomas – Port Elizabeth Technikon
Herna Viktor – University of Pretoria, South Africa
Bruce Watson – Universities of Pretoria and Eindhoven
Janet Wesson – University of Port Elizabeth

Referees

Molla Alemayehu	Klarissa Engelbrecht	Pekka Pihlajasaari
Trish Alexander	David Forsyth	Nelisha Pillay
Adi Attar	John Galletly	Laurette Pretorius
Bob Baber	Vashti Galpin	Karen Renaud
Andries Barnard	Wayne Goddard	Ingrid Rewitzky
John Barrow	Alexandré Hardy	Sheila Rock
Judy Bishop	Scott Hazelhurst	Markus Roggenbach
Gordon Blair	Johannes Heidema	Ian Sanders
Arina Britz	Tersia Hörne	Justin Schoeman
Andy Bytheway	Chris Johnson	Martie Schoeman
André Calitz	Bob Jolliffe	Elsje Scott
Charmain Cilliers	Paula Kotzé	Derek Smith
Elsabe Cloete	Derrick Kourie	Elmé Smith
Gordon Cooper	Les Labuschagne	Adrie Stander
Richard Cooper	Paul Licker	Harold Thimbleby
Annemieke Craig	Philip Machanick	Theda Thomas
Thad Crews	Anthony Maeder	Judy Van Biljon
Quintin Cutts	David Manlove	Alta Van der Merwe
Michael Dales	Gary Marsden	André van der Poll
Carina de Villiers	Thomas Meyer	Tobias Van Dyk
Alan Dix	Elsa Naudé	Lynette van Zijl
Dunlop Mark	Martin Olivier	Lucas Venter
Elize Ehlers	Don Petkov	Herna Viktor
Jan Eloff		Bruce Watson
Andries Engelbrecht		Janet Wesson

Conference

Sponsors



Keynote Abstracts

Computer Security: Hacking Tendencies, Criteria And Solutions

Martin Botha^a

Rossouw von Solms^b

Department of Information Technology, Port Elizabeth Technikon

^abothamar@saps.org.za, ^brossouw@petech.ac.za

Abstract *Computer crime and more particularly computer hacking has become increasingly active in today's business environment. Proof of this statement is a survey completed by the Computer Security Institute and the FBI which revealed that corporations, banks and governments all face a growing threat from computer crime (Berst, 1999, p1). Different methods can be used to control access to computer networks such as firewalls, but none is hacker-proof. New ways and means must therefore be defined which will minimise or eliminate computer crime. These ways should involve the utilisation of audit logs and user profiles in a proactive sense. Typical proactive actions that can be defined include: online monitoring, template analysing, generation of reports and generation of alert signals. The objective of this paper is to define and describe a proactive model which will identify a hacking attempt before it has been performed, on any computer system with more effective and easy to use graphical interfaces. This model should also provide useful tools for the security officer. It will inform the officer of different levels of hacking attempts according to statistical predefined norms.*

Keywords: *Computer security, Computer hacking, Statistical solution*

1. Introduction

"Information is the lifeblood of any organisation" (Peppard, 1993, p5). Organisations are always seeking new opportunities to utilise this information to increase their sales, by looking at ways at giving them a competitive edge over rival organisations. Some opportunities include making their services available on the Internet or to making use of electronic commerce technology.

Organisations making use of these opportunities will as a result open their doors to outsiders.

Outsiders consist of old and new customers, as well as criminals. Thus, no organisation can today function without some protection against criminals. Measures need to be devised to provide protection of all information and associated resources. Security controls (eg. user authentication, logical access control, audit logs, user profile) are such measures that can be used to provide protection for this valuable information that is found on computer systems within an organisation.

To introduce and maintain the protection measures to secure the information assets of an organisation has become an important and intricate task. For this reason a model must be

developed which will identify and inform the information security officer. The model must not only inform the officer of the different level of attempted hacking stages, but it must also provide the officer with guidelines to control and solve hacking attempts. This model needs some input from the operating system which consists of audit logs, user profiles and template analytical processes.

Audit logs keep a record of most activities and events that occur on a computer system. They exist on numerous, if not all, computing platforms such as UNIX, Windows NT, and Firewalls. Thus, seeing that the audit logs keep conscientious records of all activities and events taking place on the computer system, they should be very helpful in monitoring the state of information security within the organisation.

The second component of the model is user profiles. A user profile is a file that contains information about a user's desktop operating environment. User profiles are one of the most powerful methods available to an administrator for managing user environments. Again, numerous, if not all computing platforms make use of a user profile. The purpose of a user profile component is to provide the model with information on all authenticated users of the system. This information will then provide integrity to the model.

The third component is the template. The template consists of rules based on pre-defined criteria. This template will be placed over the audit log and the user profile to detect illegal operations (possible hacking attempt).

Today, many different ways exist for accessing an organisation's information assets and therefore all possible hacking methods must be investigated to ensure that the model will succeed. The first section of this paper will discuss the issues involved in the hacking attempt. Issues include new hacker tendencies and hacking methods.

The second section will discuss issues involved in the development of a model aimed at answering the hacking problem. Issues include the different security solutions, a proactive hacking identification model and the practical implementation of the model.

2. Hacking issues

Hacking has become a serious white collar crime and many different hacking methods exist which make the identification process very difficult. Before any identification process can be developed, all hacking issues must be investigated. This section of the paper will provide basic background on most of these issues.

2.1 New hacker tendencies

Today most corporations have more to fear from a disgruntled employee than an external intruder. But as more companies move to e-commerce, that balance may change.

According to CAI director, Patrice Papalus, the levels of computer crime are rising due to usually bored teenage computer hackers, looking for a bit of excitement. Papalus also said: "It's not simply teenagers coming in and spray painting on a WEB page and it's not just the stereotypical hacker. People are seeing financial losses due to various different kinds of attacks from professionals" (Reuters, 1999, p2).

2.2 New hacking objectives

Hacker's objectives have changed over the last few years and they can now be divided into four main categories, namely:

- Hackers that want to perform random acts of violence, or want to become famous;

- Hackers that have a personal score to settle with someone or some organisation;
- Hackers that plan to get rich by stealing information on the electronic frontier and hack only for the financial rewards; and
- Hackers who still hack for the classic objective. Classic objective referring to hackers who just want to learn how a system operates and who hack for the thrill and excitement (Donald, 1997, p17).

2.3 Hacker tools

There is a wide variety of hacker tools. One of the most useful is information gathering. The organisation's computer system will provide some information to the hacker and the other needed information will be provided by the users.

Information can be collected in five different ways. They are:

- gathering information from people;
- gathering information when going on-site;
- gathering information from the computer;
- gathering information from experts; and
- gathering information from other hackers (Donald, 1997, p19).

All these five gathering methods will provide secret information, such as login names and passwords.

2.4 Hacker access routes

Hackers access information systems mostly by way of dial-up access, dial-back systems, direct connect terminals, UUCP, network terminal / modem servers, dial-up SLIP / PPP servers and SMTP.

Dial-up access is defined as trying to guess logins and passwords. It is a most dangerous and unproductive way for a hacker to gain access to a system (Donald, 1997, p28).

A dial-back system is software that is added to a modem part, that when you call the modem line, the software will ask you for some information to authenticate the user. It will then hang up the telephone line and call the user back (Donald, 1997, p29).

Network terminal or modem servers are devices that are directly attached to the network and allow for either direct-connected terminals or modem access. In either connection cases, the remote computer will see

the user as a simple terminal connection over the network.

SLIP / PPP servers are used to extend companies networks to users who work on the road or at home. This is usually done by having a dial-up SLIP or PPP server. The server gives TCP / IP connection to a company network (Donald, 1997, p33).

2.5 Hacking techniques (methods)

Hacking techniques for hacking a system always involve password cracking. Passwords are most computer system's primary method of authentication. A password is stored in a password file.

Password crackers (hackers) need information from password files. Thus, a password cracker will use all the information available about the user, trying user's name, initials, account name, and any other personal information known. This information will be gathered from the GECOS field and from files in the user's home directory.

Most password crackers will try to perform a dictionary search. The dictionary will be based on the experiences of hackers and the knowledge of the system being attacked. The dictionary includes common first names, characters, titles, computer games and sports terms based on the industry in which the computer is being used (Donald, 1997, p39).

After a hacker gains basic privileges to a system by using password cracking or any information gathering methods, he will then look for more advanced privileges. A common method used by hackers to gain more privileges to a system, is by means of a valid password found in bad login files.

3. Hacking prevention issues

In the previous section of this paper, various methods for hackers to gain access to networks as well as methods to improve their privileges were discussed. In this section, different computer security solutions and the pro-active model will be discussed.

3.1 Computer security solutions

There are five main types of computer security solutions and all of them can be used to control hacking. The five types are as follows:

- physical security;

- document security;
- personnel security;
- hardware security; and
- software security (Smith, 1989, p54).

Physical, document and personnel security can help control hacking, but it won't halt hackers. Hardware solutions can also be used, but they have various disadvantages such as extra network traffic and expensive equipment. They can also be easily disabled by hackers without anyone noticing.

Software security provides a useful tool which can be used to successfully control and halt hackers. Software security can be set up to monitor and record the use of an information system. An unbroken chronological journal called an audit trail can be maintained of all programs, files and data records to which access, or attempted access, has been made for any job or transaction.

In most operating systems, audit trails will selectively audit the system use, discarding records that fall within the acceptable norms and making note only of unusual actions or activity. Unusual actions or activities such as an abnormally high number of incorrect log-on attempts by a user, or the unauthorized use of a peripheral such as a printer will be documented in an audit system (Smith, 1989, p117).

3.2 Audit logs

In the introduction, audit logs were briefly introduced. Audit logs form an important part of the model and two main types of audit records are used. They are:

- Keystroke monitoring; and
- Event orientated log (NIST, 1995, p.214).

Keystroke monitoring is usually considered to be a special type or case of audit log. Keystroke monitoring is used to record or view both the keystrokes entered by a user and the computer's response during a session with the user. Keystroke monitoring is used in the model to discover exactly what input the intruder typed.

Event orientated logs provide summarised information on the system uses. Audit events are comprised of three major types of audit logs. They are:

- system level audit logs;
- application level audit logs; and
- user audit logs (NIST, 1995, p.214-217).

The model uses system level audit logs to monitor and detect possible hacking attempts. As soon as a possible hacking attempt is detected, the user level audit log will be used to monitor the user activity on the system. The application level audit log is also used by the model to monitor the user's activities in an application as soon as a trend has been found.

3.3 User profile

The user profile has also been briefly introduced in the introduction. A user profile is a file that contains information about a user's desktop operating environment (Block, 1994, p55). Almost all computing platforms such as UNIX and Windows NT provide user profile input.

The model transforms these user profile inputs to create its own user profile database. The database information is then used by the processing component to make decisions which will provide integrity to the model.

An example of user profile database information includes:

- user rights and privileges; and
- user normal working hours.

3.4 Template

The template is a very complex, difficult and resource intensive component. This component is based on trend analysis and consists of hundreds and hundreds of rules. The rules included in the template are based on statistical norms. These norms are used to determine whether a certain level of confidence is reached. If so, the program will claim that the detected process is a possible hacking attempt (security violation).

Some of the rules include:

Rule 1:

If the model traces an invalid password attempt in the system audit log for the first time, the probability of a hacking activity is equal to one percent.

Rule 3:

If the model traces an invalid password attempt in the system audit log for the third time, and the user profile classifies the user as a low class user, the probability of a hacking activity is equal to four percent.

Three values are defined as important values in the template. They are:

- MIP;
- MP; and
- MAP.

MIP (minimum probability of hacking activity) is the value at which the model will send its first alert message to the security administrator. It sends the message before a hacker starts with his/her main hacking activities.

MP (medium probability of hacking activity) is the value at which the second alert message is sent to the security administrator. The message includes details of the security violation presently in operation.

MAP (maximum probability of hacking activity) is the value at which the model will send its final alert message to the security administrator. This message states that a security violation has occurred.

3.5 The pro-active identification model

The pro-active identification model is developed according to the system method. The audit logs, user profile and template form the basis for this model. The model has three features, namely:

- input section;
- processing section; and
- output section.

Figure A.1 show the components of the model. The figure also shows that the model is separated from the traditional audit system and runs in parallel with it.

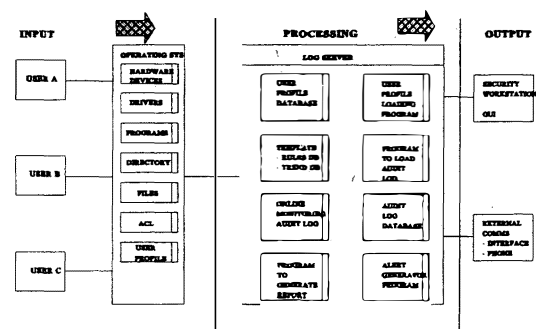


FIGURE A.1 : GENERAL REPRESENTATION OF THE MODEL

The input section consists of users and operating systems. The operating systems provide information to the user profile loading and building program. The operating systems also

provide information on the system level logs, application level logs and user audit logs.

The processing section consists of ten independent elements. See elements in figure A.2.

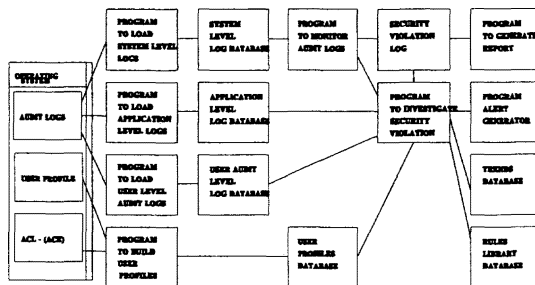


FIGURE A.2 : GENERAL REPRESENTATION OF THE EXECUTION OF THE MODEL

The first step in the processing execution is to load all user profiles from the operating system. The loading and building of the user profiles are performed periodically and the execution of it is done by any read variable script. Any read variable script can also be used to load and store system level, application level, and user audit level logs into their various databases.

The program security violation investigation module (PSVIM) forms the heart of the model. PSVIM is a program which is normally written in a procedural language such as C++. The program will be triggered by a trigger code generated by the program audit log monitor module.

The trigger code consists of two parts. The first part contains the trigger code instruction which triggers the module, and the second part is the code and sequence number of the violation. The PSVIM program will then copy the details of the violation in the security violation log module, and it also transfer the appropriate rules set in the rule library databases. Lastly the program also loads the corresponding user profiles from the user profile databases.

The PSVIM program will then start to execute the rules onto the audit logs input and user profiles. The program can also extract information out of the application level log databases or user audit log databases according to the security violation. Execution of these rules is performed similarly to the execution of the rules explained in the template section.

The program report generator module is also part of the processing section. This module performs two operations. The first is to generate a report on the security violation. The second operation is to execute a interface operation between the model and the Graphics User Interface module on the security officer working station.

The last of the mentioned processing elements is the program alert or generator module. This module performs alert interface functions such as:

- E-mail alert – services;
- Cellular phone alert calls; and
- Conventional phone alert calls.

The output section consists of two elements. The elements are:

- Security working station; and
- External communication module.

The security working station is a computer that is used by the information security officer to interact with the program security violation investigation module. The security working station is an intuitive GUI, which is used to control most of the functions associated with the program security violation investigation module.

The external communication module consists of E-mail interface and an electronic communication interface phone network.

3.6 Advantages

The pro-active identification model can only improve the information security of an organisation. The model has these inherent advantages:

- *Platform independence:* The model can be developed to execute in any network operating system environment such as UNIX and Windows NT.
- *Transparent audit log analysis:* The model provides transparent audit log analysis. This is achieved because the security administrator does not have to know the location of the different audit logs. The GUI uses easy markup language which will fetch the log files for the security administrator.
- *Scalability:* The operating system will provide the model with the information on user profiles and audit log input. This enables the model to deploy if new users and applications are inserted on the system. Thus no changes to the model are

needed to achieve scalable deployment on the system.

- *Consistency*: The consistency feature is one of the main advantages of this model. Consistency refers to the ability of the model to provide the same conclusion if the same security violations occur repeatedly. This feature is made possible due to the rules library and the user profile.
- *Security analysis is independent from operating system*: The model performs security analysis independently from the operating system. This means that if the hacker tries to cover his tracks by changing the operating audit log files, he will still be detected by the model, since it stores its own copy of the audit log.
- *Easy graphic user interface*: The model provides easy-to-use GUI for the security administrator. The GUI is a commonly known product and it reduces training costs for security administrators.

3.7 Implementing the model practically

The purpose of the practical implementation is to prove that the theoretical model is feasible. An expert system shell was used to develop the prototype of the model to prove this statement.

CLIPS expert system language was chosen to develop the model. CLIPS is a forward chaining rule-based language that has inferencing and representation capabilities similar to those of OPS 5 (Glarratano J, 1989, p373). CLIPS was written in C which is intended to serve as a versatile tool for the development and implementation of expert systems.

The development of the prototype consists of three main elements. They are:

- fact list: global memory for data;
- knowledge-base: contains all the rules; and
- inference engine: controls overall execution.

A fact-list consists of facts. A "chunk" of information in CLIPS is called a fact. The CLIPS program matches the facts of the left and right parentheses. Examples of facts include:

- Invalid;
- Times; and
- User

Knowledge-base contains all the rules needed to execute the model. Some of the rules include:

RULE: To determine the file information

IF: the phase is to be determined if the password is invalid

THEN: determine how many times the invalid password occurs, and

IF: invalid password is equal to or more than six times, fire rule "determine user rights", else monitor and log.

CF: 90

An inference engine is a program that makes decisions and judgements, using search and heuristic reasoning procedures based upon symbolic data contained in the knowledge base (Maus R, 1990, p14). Reasoning is the process of drawing inferences from known facts, and an inference is the logical conclusion based on available information (Badiru B, 1991, p106).

The prototype uses a mixture of forward and backward reasoning strategies.

3.8 Testing results

Testing makes use of six input files which simulate the input received and generated by the model. It also uses four output files which the prototype uses to save its output.

The prototype was executed, and all test messages and executions on input data were performed correctly. Although the execution was performed at a low cost, multiple expertise and with fast response, two major drawbacks were encountered. Limited scope and limited facts of representation were encountered due to the use of the expert system shell which language was not procedural such as C++.

Considering all the advantages and disadvantages of the prototype, it was clear that the advantages provided by the prototype, outweighed the disadvantages. Thus, the prototype proved that the pro-active model as discussed in the previous section is feasible and with future enhancement it can be developed in a revolutionary method to control security violations.

3.9 Further research

The research done during this mini-thesis proved that a model can be developed to identify a hacking activity before, during or after the performance. Further research must still be performed in hacker's behaviour and

hacking methods. This research will improve the accuracy of the model.

Future research must also include:

- Investigation of means to ensure reliable backup mechanisms to handle interruptions;
- Investigation of means to ensure proper security for the model; and
- Investigation of means to reduce downgrading of system performance.

4. Conclusion

This document provides a new exciting method to stop and control computer crimes (hacking). Hacking will become a bigger problem in the future and many new methods will be developed to prevent it. One positive aspect of this research is that parts of the complete model can be used in the future to solve the hacking problem.

The model discussed in this document was found to be user-friendly and effective in detecting security violations.

Although the model is effective, it must never be used to replace the responsibility of the security officer, but it must assist him/her in the protection of the organisation information and associated resources.

Organisations will only SURVIVE in the twenty-first century, if there is a business reason as well as secure security mechanisms which will ensure that the business needs are executed securely.

5. References

- [1] Berst, J. (1999). Hacking 101 : Why should you know what they know [online]. Zdnet. Available from Internet : URL http://www.zdnet.com/anchordesk/story/story_3163.html
- [2] Block, B. (1994). WINDOWS NT 3.5 - GUIDELINES FOR SECURITY, AUDIT AND CONTROL (1st ed.). Canada : Canada Publishing Corporation.
- [3] Cornwall, H. (1989). THE HACKERS HANDBOOK III(3rd ed.). London : Century Hutchison LTD.
- [4] Donald, L. (1997). HALTING THE HACKERS(1st ed.). London : Hali PTR.
- [5] Giarratano, J. (1989). Expert Systems (1st ed.). Boston, Massachusetts, U.S.A. : PWS-KENT publishing Company.
- [6] Hoffner, K.& Markoff, J. (1991). Cyberpunk [online]. Touchstone. Available from Internet : URL <http://www.comm.fsu.edu/com4330/summer/essay/lrp1.htm>
- [7] Neil, S. (1999). Beware the crackers, they're out in full force. [online]. Available from the Internet : URL <http://www.zdnet.com/pcweek/stories/columns/0,4351,1013786,00.html>
- [8] Peppard, J. (1993). Information, technology and strategy. In J. Peppard (Ed.). I.T. strategy for business (pp. 1-25). London: Pitman Publishing.
- [9] Pietrucha, B. (1998 March 20). Newsbytes. Teen hackers giving way to move to more serious threats. In Computer Select [CD-ROM]. Foster City, Calif. Information Access Company.
- [10] Ramkrishna, S. (1987). UNIX UTILITIES(1st ed.). Berkshire: McGraw-Hill.
- [11] Schwartz, R.L. & Wall, L. (1991). Programming Perl. U.S.A.: O'Reilly & Associates, Inc.
- [12] Smith, M.R. (1989). COMMONSENSE COMPUTER SECURITY (1st ed.). Berkshire: McGraw-Hill.
- [13] Sterling, B. (1994). The Hacker crackdown - The digital underground [online]. Available from Internet : URL <http://www.usfca.edu/crackdown/crack5.htm>
- [14] US Department of Commerce. (1995). An introduction to computer security: The NIST handbook. Washington: U.S. Government Printing Office.