The South African Institute for Computer Scientists and
Information Technologists

ANNUAL RESEARCH AND DEVELOPMENT SYMPOSIUM

23-24 NOVEMBER 1998
CAPE TOWN
Van Riebeeck hotel in Gordons Bay

Hosted by the University of Cape Town in association with the CSSA,
Potchefstroom University for CHE and
The University of Natal

PROCEEDINGS

EDITED BY
D. PETKOV AND L. VENTER

SPONSORED BY

ABSA Group

The South African Institute for Computer Scientists and Information Technologists

ANNUAL RESEARCH AND DEVELOPMENT SYMPOSIUM

23-24 NOVEMBER 1998
CAPE TOWN
Van Riebeeck hotel in Gordons Bay

Hosted by the University of Cape Town in association with the CSSA,
Potchefstroom University for CHE and
The University of Natal

GENERAL CHAIR : PROF G. HATTINGH, PU CHE

PROGRAMME CO-CHAIRS:
PROF. L VENTER, PU CHE (Vaal Triangle), PROF. D. PETKOV, UN-PMB

LOCAL ORGANISING CHAIR: PROF. P. LICKER, UCT - IS

PROCEEDINGS

EDITED BY
D. PETKOV AND L. VENTER

SYMPOSIUM THEME:

Development of a quality academic CS/IS infrastraucture in South Africa

SPONSORED BY

ABSA Group

Proceedings of the 1998 Annual Research Conference of the South African Institute for Computer Scientists and Information Technologists.
Edited by Prof. D. Petkov and Prof. L. Venter
Van Reebeck Hotel, Gordons Bay, 23-24 November 1998

ISBN: 1-86840-303-3

Keywords: Computer Science, Information Systems, Software Engineering.

The views expressed in this book are those of the individual authors and not of the South African Institute for Computer Scientists and Information Technologists.

Office of SAICSIT: Prof. J.M.Hatting, Department of Computer Science and information Systems,
Potchefstroom University for CHE, Private Bag X6001, Potchefstroom, 2520, RSA.

Produced by the Library Copy Centre, University of Natal, Pietermaritzburg.

FOREWORD

The South African Institute for Computer Scientists and Information Technologists (SAICSIT) promotes the cooperation of academics and industry in the area of research and development in Computer Science, Information Systems and Technology and Software Engineering. The culmination of its activities throughout the year is the annual research symposium. This book is a collection of papers presented at the 1998 such event taking place on the 23rd and 24th of November in Gordons Bay, Cape Town. The Conference is hosted by the Department of Information Systems, University of Cape Town in cooperation with the Department of Computer Science, Potchefstroom University for CHE and and Department of Computer Science and Information Systems of the University of Natal, Pietermaritzburg.

There are a total of 46 papers. The speakers represent practitioners and academics from all the major Universities and Technikons in the country. The number of industry based authors has increased compared to previous years.

We would like to express our gratitude to the referees and the paper contributors for their hard work on the papers included in this volume. The Organising and Programme Committees would like to thank the keynote speaker, Prof M.C.Jackson, Dean, University of Lincolshire and Humberside, United Kingdom, President of the International Federation for Systems Research as well as the Computer Society of South Africa and The University of Cape Town for the cooperation as well as the management and staff of the Potchefstroom University for CHE and the University of Natal for their support and for making this event a success.

Giel Hattingh, Paul Licker, Lucas Venter and Don Petkov

Table of Contents

EFFECTIVE INFORMATION SECURITY MONITORING USING DATA LOGS

Willem Krige and Rossouw von Solms
Department of Information Technology
Port Elizabeth Technikon
Private Bag X6011
Port Elizabeth 6000
SOUTH AFRICA
E-mail: rossouw@ml.petech.ac.za

Abstract

Log files or audit logs are files that record information about events that occur on a computer system at all times, that is if the log files are configured correctly. Most information security officers do not use this resource to its fullest potential. Even though log files are kept on a computer system they can be regarded as dead data, simply because they are very seldomly used by the information security officer(s), unless a crisis arises. Thus, some information security officers do not consider log files to be of much importance and have the opinion that the log files just take up space and other valuable resources. However, log files can be used more proactively to improve the information security of an organisation. The primary objective of this project is to use this available resource in a proactive manner rather than a post-mortem (reactive) manner as it is currently being used in most organisations.

Log files occur on various different operating platforms such as UNIX, WindowsNT, etc. No standards however exist for log files, so not many of the log files are compatible with each other. Each and every vendor has there own method of creating log files, thus there is no conformity in this area. The only main similarity is that most of the log files are in ASCII format. A model is envisaged, which will allow the log files from the various operating platforms to be integrated together. Thus all log files from all different operating platforms can apply the same model, to improve information security. A prototype is currently under development, which makes use of a GUI interface (web-browser) that interfaces with the log files in real-time as well as on a non real-time basis. Various scripts are also under construction, which allow various security related events in the log files to be monitored in real-time. If an event, which is out of the ordinary is detected by the script monitoring the log file, an appropriate alert (e-mail, online message, etc.) is sent to the information security officer(s).

As the WWW (Internet) is growing at a rapid rate means that the chances of someone breaking into a computer system is continually increasing. Thus a computer system needs to have security in place to help curb this problem. There are numerous countermeasures (controls/safeguards) that are used to try and improve the security of computer systems.

Log files are an existing resource, which can be found on most, if not all computer systems. However, the problem is that this resource is not being used to its full potential. In most organisations they are only used after an incident occurs (post-mortem). The log files are thus being used to solve the problem(s) *after* they occur.

The log files do however record the events as they take place on the computer system. Thus if the log files could be monitored in real-time or virtually real-time would provide a means of detecting events as they occur or provide an early warning to the security officer concerned. If the log files could be monitored in real-time would mean that the log files would be used in a proactive manner. It would also allow for the improved utilisation of an existing resource as well as providing information security and an early warning mechanism.

The prototype is currently being implemented in a UNIX environment. Numerous scripts have been written in PERL to monitor the log files in real-time. Some of the scripts are configurable which allows the security officer to customise the scripts for the organisation's needs. The security officer can

specify what should be monitored in the log files e.g. a bad su attempt. He/she can then specify how he or she should be notified depending on the severity of the event. A message can either be sent to the security officer via email, an online message or a sms message, thus alerting him/her of an incident as it occurs and not days later when the damage has already been done. This component is commonly referred to as *online monitoring*. The incident is also recorded in a file, which can be used to provide the security officer with a summary of incidents that have occurred. This is referred to as an *exception report.*

Another major problem associated with the log files is that it is very time-consuming to manually search through the data for specific information. A GUI has been developed which can search for information in the log files in a quick an user-friendly manner e.g. If the security officer only wanted to see log entries concerned with a specific user name, or log entries about all connections from a specific host. Such searches/queries would be very simple, quick and easy to perform. This component is known as a *condition report.*

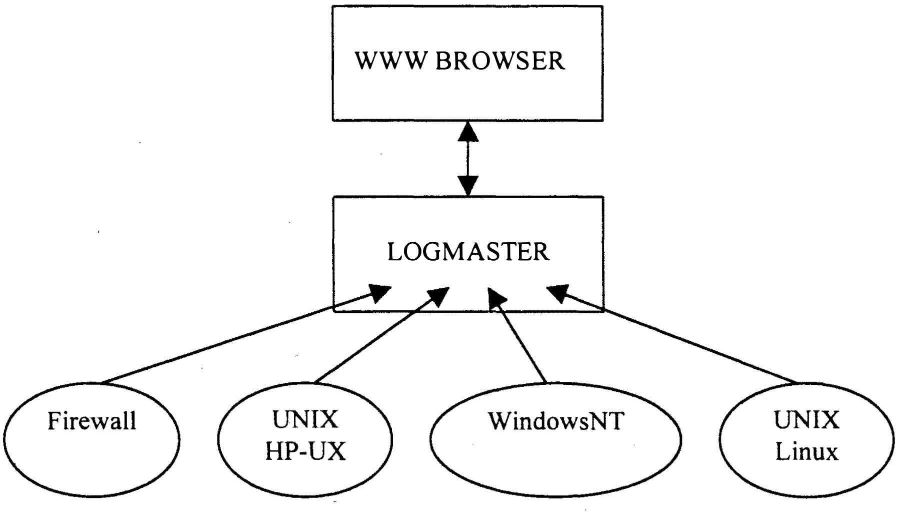
The GUI is thus very useful as it provides a graphical means of interfacing with the log files, which makes it user-friendlier. The GUI makes use of web pages and a web-browser (Netscape, etc.) to perform these searches on the log files. By making use of this method means that the security officer can access the log files from any machine that has access to the WWW without having to install any programs locally.

So far condition reports (GUI), online monitoring (PERL Scripts) and exception reports have been introduced. The last component is *trend analysis*. This component will be used to identify certain trends in the log files that cannot be detected very easily e.g. someone attempting to hack into a user account over a long period of time. This component has not yet been started.

Thus there are four major components which need to be implemented in order to use the log files in a proactive manner. These components can either be done in real-time or non real-time, depending on the component concerned. To summarise, here are the four components:

- Online Monitoring (Real-Time)
- Condition Reports (Real-Time)
- Exception Reports (Real and/or Non Real-Time)
- Trend Analysis (Non Real-Time)

If these four components are implemented, then the log files can be used more proactively and thus help improve the security of the computer system. The diagram below shows a representation of how it all fits together.



A central host (Logmaster) will be used to keep a duplicate of all the log files from the various systems. As an event occurs, the event is logged locally as well as on the Logmaster. The Logmaster will also be a web server so that the web-browser can interface with the log files via CGI (Common Gateway Interface). Online monitoring will be performed on the log files on the Logmaster in real-time. Condition and exception reports would be performed by means of the browser. The request(s) will be sent to the Logmaster and the results returned to the browser. Trend analysis needs to be done in non real-time and can be performed on the log files on the Logmaster. The Logmaster will thus reduce the burden that would be put on the host systems as well as providing a backup of the log files. It will also reduce security risks and compatibility problems encountered as each system would not have to be setup as a web server and have its scripts customised.

Log files are valuable resources, which are currently not being used to their fullest potential. The model and tools under development will hopefully help solve some or most of the problems presently associated with log files.