ALGORITHM ENHANCEMENT FOR HOST-BASED INTRUSION DETECTION SYSTEM USING DISCRIMINANT ANALYSIS

DAHLIYUSMANTO

A thesis submitted in fulfillment of the
requirements for the award of the degree of
Master of Science (Computer Science)

Faculty of Computer Science and Information System
Universiti Teknologi Malaysia

JULY 2004

ABSTRACT

Algorithms for building detection models are usually classified into two categories: *misuse detection* and *anomaly detection*. Misuse detection algorithms model know attack behavior. They compare sensor data to attack patterns learned from the training data. Anomaly detection algorithms model normal behavior. Anomaly detection models compare sensor data to normal patterns learned from the training data by using statistical method and try to detect activity that deviates from normal activity. Although Anomaly IDS might be complete, its accuracy is questionable since this approach suffers from a high false positive alarm rate and misclassification. This thesis expects an enhancement algorithm to be able to reduce a false positive alarm and misclassification rate. This research investigated a discriminant analysis method for detecting intrusions based on number of system calls during an activity on host machine. This method attempts to separate intrusions from normal activities. This research detects intrusions by analyzing at least system call occurring on activities, and can also tell whether an activity is an intrusion. The focus of this analysis is on original observations that performed a detecting outlier and power transformation to transform not normally distributed data to near normality. The correlation of each system calls are examined using coefficient correlations of each selected system call variables. This approach is a lightweight intrusion detection method, given that requires only nine system calls that are strongly correlated to intrusions for analysis. Moreover, this approach does not require user profiles or a user activity database in order to detect intrusions. Lastly, this method can reduce a high false positive alarm rate and misclassification for detecting process.

ABSTRAK

Algoritma untuk membina model pengesanan pada lazimnya dikelaskan kedalam dua kategori: pengesanan penyalahgunaan dan pengesanan kelainan. Algoritma pengesanan penyalahgunaan memodelkan kelakuan serangan yang telah diketahui. Ia membandingkan data pengesan dengan corak-corak serangan yang telah diketahui dari data latihan. Algoritma pengesanan kelainan memodelkan kelakuan biasa. Model pengesanan kelainan ini membandingkan data sensor dengan corak-corak biasa yang telah dipelajari dan mencuba mengesan aktiviti yang menyimpang dari aktiviti-aktiviti biasa. Walaupun kaedah pengesanan kelainan boleh lengkap, namun ketepatanya dapat dipersoalkan selepas pendekatan ini mendapati sebuah kadar bunyi amaran palsu yang tinggi dan kadar kesalahan mengklasifikasi. Tesis ini menjangkakan sebuah perbaikan algoritma yang boleh mengurangkan kadar bunyi amaran palsu dan kadar kesalahan mengklasifikasi. Penyelidikan ini mengkaji satu kaedah analisis diskriminant untuk mengesan pencerobohan berdasarkan bilangan “system call” semasa aktiviti keatas mesin hos dilaksanakan. Kaedah ini mencuba untuk mengasingkan pencerobohan-pencerobohan daripada aktiviti-aktiviti biasa. Hasil penyelidikan ini mengesan pencerobohan dengan menganalisis sekurang-kurangnya “system call” yang berlaku keatas aktiviti-aktiviti, dan juga berkeupayaan memberitahu samada aktiviti tersebut merupakan satu pencerobohan. Analisis ini memfokuskan kepada pemerhatian asal yang melaksanakan pengesanan terpencil dan transformasi kuasa untuk menukarkan data teragih yang tidak biasa kepada pembiasaan yang terhampir. Korelasi setiap “system call” dikaji menggunakan korelasi koeffisien bagi setiap pembolehubah “system call” yang dipilih. Pendekatan ini adalah kaedah pengesanan pencerobohan yang mudah, memandangkan analisa ini hanya memerlukan sembilan system call yang mempunyai korelasi yang kuat terhadap pencerobohan. Tambahan lagi, kaedah ini tidak memerlukan profil pengguna ataupun pangkalan data aktiviti-aktiviti pengguna untuk mengesan pencerobohan. Akhir sekali, kaedah ini dapat mengurangkan kesalahan mengklasifikasi dalam proses pengesanan.

TABLE OF CONTENTS

4 Preliminary Analysis

5 Conclusion and Discussions

LIST OF TABLES

LIST OF FIGURES

ABREVIATIONS

AER	-	Actual Error Rate
APER	-	Apparent Error Rate
BSM	-	Basic Security Module
CPU	-	Central Processing Unit
DoS	-	Denial of Service
ECM	-	Expected Cost of Misclassification
IDES	-	Intrusion Detection Expert System
IDS	-	Intrusion Detection System
IPA	-	Information-technology Promotion Agency
IRS	-	Internet Response System
LAN	-	Local Area Network
LBSM	-	Linux Basic Security Module
MIDAS	-	Multics Intrusion Detection & Alerting System
MIT	-	Massachusetts Institute of Technology
NATE	-	Network Analysis of Anomalous Traffic Events
NIDES	-	Next Generation Real-time Intrusion Detection Expert System
NSM	-	Network Security Module
PCA		Principal Component Analysis
pH	-	Process Homestasis
SNARE	-	System Intrusion Analysis & Reporting Environment
SSO	-	Senior Security Officer
TCP	-	Transmission Control Protocol
UNM	-	University of New Mexico

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Overview

Attacks on computer infrastructures are a serious problem. Over the past twelve years, the growing number of computer security incidents on the Internet has reflected the growth of the Internet itself. Because most deployed computer systems are vulnerable to attack, intrusion detection is a rapidly developing field. Intrusion detection is an important technology business sector as well as an active area of research (Allen *et al.*, 2000).

There are many reasons why a computer system behaves in an undesired way. For a problem to be categorized as a security problem it must in some ways involve the fact or possibility that a human being does something that is not permissible. It is normally the person or organization who owns the system and/or the information who decides what is allowed and what is not. Wrongdoors can be categorized as *insiders* or *outsiders*. Insiders are persons related to the owner organization who try to misuse or extend their privileges. Outsiders are attackers who are unrelated to the owner organization who try to gain entry to systems (Cheswick, 1992). Within the community of security officers and researchers, insiders threat is considered much more dangerous than the threat from outsiders, but the media have conveyed the opposite picture to the general public.

The security of a computer system is compromised when an intrusion takes place. An intrusion can be defined as any set of actions that attempt to compromise the integrity,

confidentiality or availability of a resource (Heady *et al.*, 1990). There are prevention techniques, such as user authentication (e.g. using passwords or biometrics), avoiding programming errors, and information protection (e.g., encryption) have been used to protect computer systems as a first line of defense. These techniques alone is not sufficient because as systems become ever more complex, there are always exploitable weaknesses in the systems due to design and programming errors, or various "socially engineered" penetration techniques. The policies that balance convenience versus strict control of a system and information access also make it impossible for an operational system to be completely secure.

1.2 Background of Problems

Intrusion detection has been an active field of research for the last two decades. This is exemplified by an influential paper, published in 1980, "Computer Security Threat Monitoring and Surveillance" by James Anderson (1980). It was followed some years later by the seminal paper "An Intrusion Detection Model" by Denning (1987). The author also introduced the concept of an Intrusion Detection System (IDS) as a second line of defense. The paper provides a methodological framework that inspires many researchers and, in more recent times, laid the groundwork for commercial products. In order to detect intrusion, two critical errors can be generated by IDS namely false positive and false negative error. A false negative error is intrusive behavior defined by the IDS as normal user behavior while the false positive error is legitimate user behavior that is regarded by the IDS as intrusive behaviors.

Intrusion Detection System (IDS) can be categorized as network-based or host-based. In the former, header fields of the various network protocols are use to detect intrusions. In the later approach (host-based IDS), the focus shifts to the operating system level. System call data is extracted from audit logs like the Solaris Basic Security Modules (BSM) and Linux Basic Security Modules (LBSM) and their behavior is studied to detect attacks.

Two different lines of approach have been adopted to detect intrusions. The first technique is *misuse (signature) detection*, this technique similar to pattern matching. Systems are modeled upon known attack patterns and the test data is checked for the occurrence of these patterns. Theses systems have a high degree of accuracy but fail to detect new attacks. For this reason, the knowledge base needs to be updated regularly in order to add new intrusion scenarios (Javitz and Valdes, 1993). This updating has to be done by experts or the designers of the systems. It is done manually and puts an extra workload on the security administrator (Campbell *et al*., 2003). The second technique is *anomaly detection*, a detection method that finds intrusions by analyzing the deviation from normal activities (Frank, 1994) and usually at the user level or system level (Sundaram, 2001).

In general, most anomaly detection system learns a normal system activity profile, and then flags all system events that statistically deviate from the established profile. One of the strength of anomaly detection is the ability to abstract information about the normal behavior of a system and detect attacks regardless of whether of not the system has seen them before. The primary advantage of this approach is that it can detect unknown attacks. Most behavior models are built using metrics that are derived from system measures such as CPU usage, memory usage, number and time of logins, and network activity. However it creates very large overhead for the host machine (Spafford, 1995), which must have the capacity to record all users' activities in a database, create users' profiles database (Lundin and Jonsson, 2000) based on defined measures for intrusion detection (Denning, 1987), and sometimes the sequences of data are not independent or overlap. And next, the weakness of anomaly detection system can generate a lot of false alarms (Tandon and Chan, 2003). This is attributed to the fact that not all anomalies are necessarily attacks and will thus result in false positive. It also has the vulnerability to an intruder who breaches the system during their learning phase (Endler, 1998). A savvy intruder can gradually train the anomaly detector to interpret intrusive events as normal system behavior.

Most of the present techniques for host-based anomaly detection systems revolve around system call. System call traces are a common type of audit data collected for performing intrusion detection. There are many researches (Forrest *et al.*, 1999; Provost

(2003), Somayaji *et al*. (2000), Eskin *et al*. (2001) which system calls are used to characterize normal behavior programs, each of which involve building or training a model using traces of normal processes (Carrasco and Oncina, 1994).

Many researchers have been statistical analysis to detect intrusions. Vaccaro and liepins (1989) used statistic to monitor changes in user behavior, Heberlein (1990) used statistical along with rules to monitor LAN traffict, Neuman and Porras (1999) statistical component was in herited from SRI IDS and Chapple (2000) was developed a classification tree approach to formulate statistically derived rule set for classifying intrusive activity.

There are some statistical approaches in order to detect intrusion. For example, statistical threshold detection approaches. The goal of this approach is to record each occurrence of that specific event, as the name implies, detecting when the number occurrences of that event exceed a reasonable amount that one might expect to occur during normal system operation. Disadvantage of this approach is a poor detector of even semi-sophisticated intrusions. Other approach is statistical profile-based. This approach is based on the assumption that violations involve abnormal use of the system. The main advantage of the statistical profile based detection approach is does not require any prior knowledge of the target system. However, the approach is very difficult to determine threshold which an anomaly should be considered intrusive.

Statistical method also used to analyzes system calls in privileged processes with discriminant analysis with Mahalanobis distance (Midori *et al*., 2001). This method appears to be quite efficient utilizing only eleven system calls to discriminate between normal and intrusive activities. Moreover, the approach does not require user profiles or a user activity database in order to detect intrusions. However, their method has a problem; the result of analysis for all samples in their method still have high misclassification in order to discriminate between intrusion and normal activities.

1.3 Research Objectives

The primary objectives of this study are; 1) to separate intrusion from normal activities using appropriate system call based on discriminant analysis method, and 2) to reduce the probability of misclassification during detecting process. These are followed by the second objectives:

i) To develop IDS data transformation.
ii) To develop suitable model for IDS testing.
iii) To propsoe an enhancement algorithm to improve IDS misclassification rate.

1.4 Research Scope and Limitations

This study focuses on improvement of intrusion detection in statistical approach. A new algorithm is to be developed for discriminant analysis method using data set of system calls executed by active process. This data set contains programs that run as daemon program and console command programs, and different kinds of intrusion such as buffer overflow, symbolic link attack and Trojan programs. This study only those programs that run with privilege, because misuse of these programs has the greatest potential for harm to the system. Finally, the study will close by presenting an overall intrusion detection using disecriminant analysis method in minimizing false alarm for detection.

It is widely accepted that one of the most important characteristic about IDS is that they must correctly identify intrusions and attacks. Furthermore, there are really only two possible decisions for each activity that IDS observe: the activity can be positively identify as an attack or as benign. Therefore, this study is mainly concerned with intrusion detection using discriminant analysis method in order to correctly identify intrusive and normal activities.

1.5 Research Contributions

In this thesis, data set of system call are used for measuring the performance of the observed and developed algorithm. The selection of appropriate system call will deliver applicable techniques for the intrusion detection. However, there is a point contributed by this thesis :

"*A new algorithm to reduce misclassification for detecting process, to increase detection rate and to minimize false alarm*."

1.6 Organization of Research

This study is divided into six chapters. The remaining chapters are organized as follows. Chapter 2 presents a literature review and overview of intrusion detection, system calls, statistical method, and the previous studies regarding the topics interest. Chapter 3 discusses the research method and data characteristics to be used in the present study. Chapter 4 presents extracting process detecting outlier's outputs, and transformation data. Next, this chapter also describes discriminant analysis results. Finally, chapter 5 offers a review of the objectives and to make conclusions and discussions as well, and some directions for future research.

TABLE OF CONTENTS

4 Preliminary Analysis

5 Conclusion and Discussions

LIST OF TABLES

LIST OF FIGURES

ABREVIATIONS

AER	-	Actual Error Rate
APER	-	Apparent Error Rate
BSM	-	Basic Security Module
CPU	-	Central Processing Unit
DoS	-	Denial of Service
ECM	-	Expected Cost of Misclassification
IDES	-	Intrusion Detection Expert System
IDS	-	Intrusion Detection System
IPA	-	Information-technology Promotion Agency
IRS	-	Internet Response System
LAN	-	Local Area Network
LBSM	-	Linux Basic Security Module
MIDAS	-	Multics Intrusion Detection & Alerting System
MIT	-	Massachusetts Institute of Technology
NATE	-	Network Analysis of Anomalous Traffic Events
NIDES	-	Next Generation Real-time Intrusion Detection Expert System
NSM	-	Network Security Module
PCA		Principal Component Analysis
pH	-	Process Homestasis
SNARE	-	System Intrusion Analysis & Reporting Environment
SSO	-	Senior Security Officer
TCP	-	Transmission Control Protocol
UNM	-	University of New Mexico

LIST OF APPENDICES

confidentiality or availability of a resource (Heady *et al.*, 1990). There are prevention techniques, such as user authentication (e.g. using passwords or biometrics), avoiding programming errors, and information protection (e.g., encryption) have been used to protect computer systems as a first line of defense. These techniques alone is not sufficient because as systems become ever more complex, there are always exploitable weaknesses in the systems due to design and programming errors, or various "socially engineered" penetration techniques. The policies that balance convenience versus strict control of a system and information access also make it impossible for an operational system to be completely secure.

1.2 Background of Problems

Intrusion detection has been an active field of research for the last two decades. This is exemplified by an influential paper, published in 1980, "Computer Security Threat Monitoring and Surveillance" by James Anderson (1980). It was followed some years later by the seminal paper "An Intrusion Detection Model" by Denning (1987). The author also introduced the concept of an Intrusion Detection System (IDS) as a second line of defense. The paper provides a methodological framework that inspires many researchers and, in more recent times, laid the groundwork for commercial products. In order to detect intrusion, two critical errors can be generated by IDS namely false positive and false negative error. A false negative error is intrusive behavior defined by the IDS as normal user behavior while the false positive error is legitimate user behavior that is regarded by the IDS as intrusive behaviors.

Intrusion Detection System (IDS) can be categorized as network-based or host-based. In the former, header fields of the various network protocols are use to detect intrusions. In the later approach (host-based IDS), the focus shifts to the operating system level. System call data is extracted from audit logs like the Solaris Basic Security Modules (BSM) and Linux Basic Security Modules (LBSM) and their behavior is studied to detect attacks.

Two different lines of approach have been adopted to detect intrusions. The first technique is *misuse (signature) detection*, this technique similar to pattern matching. Systems are modeled upon known attack patterns and the test data is checked for the occurrence of these patterns. Theses systems have a high degree of accuracy but fail to detect new attacks. For this reason, the knowledge base needs to be updated regularly in order to add new intrusion scenarios (Javitz and Valdes, 1993). This updating has to be done by experts or the designers of the systems. It is done manually and puts an extra workload on the security administrator (Campbell *et al*., 2003). The second technique is *anomaly detection*, a detection method that finds intrusions by analyzing the deviation from normal activities (Frank, 1994) and usually at the user level or system level (Sundaram, 2001).

In general, most anomaly detection system learns a normal system activity profile, and then flags all system events that statistically deviate from the established profile. One of the strength of anomaly detection is the ability to abstract information about the normal behavior of a system and detect attacks regardless of whether of not the system has seen them before. The primary advantage of this approach is that it can detect unknown attacks. Most behavior models are built using metrics that are derived from system measures such as CPU usage, memory usage, number and time of logins, and network activity. However it creates very large overhead for the host machine (Spafford, 1995), which must have the capacity to record all users' activities in a database, create users' profiles database (Lundin and Jonsson, 2000) based on defined measures for intrusion detection (Denning, 1987), and sometimes the sequences of data are not independent or overlap. And next, the weakness of anomaly detection system can generate a lot of false alarms (Tandon and Chan, 2003). This is attributed to the fact that not all anomalies are necessarily attacks and will thus result in false positive. It also has the vulnerability to an intruder who breaches the system during their learning phase (Endler, 1998). A savvy intruder can gradually train the anomaly detector to interpret intrusive events as normal system behavior.

Most of the present techniques for host-based anomaly detection systems revolve around system call. System call traces are a common type of audit data collected for performing intrusion detection. There are many researches (Forrest *et al.*, 1999; Provost

(2003), Somayaji *et al*. (2000), Eskin *et al*. (2001) which system calls are used to characterize normal behavior programs, each of which involve building or training a model using traces of normal processes (Carrasco and Oncina, 1994).

Many researchers have been statistical analysis to detect intrusions. Vaccaro and liepins (1989) used statistic to monitor changes in user behavior, Heberlein (1990) used statistical along with rules to monitor LAN traffict, Neuman and Porras (1999) statistical component was in herited from SRI IDS and Chapple (2000) was developed a classification tree approach to formulate statistically derived rule set for classifying intrusive activity.

There are some statistical approaches in order to detect intrusion. For example, statistical threshold detection approaches. The goal of this approach is to record each occurrence of that specific event, as the name implies, detecting when the number occurrences of that event exceed a reasonable amount that one might expect to occur during normal system operation. Disadvantage of this approach is a poor detector of even semi-sophisticated intrusions. Other approach is statistical profile-based. This approach is based on the assumption that violations involve abnormal use of the system. The main advantage of the statistical profile based detection approach is does not require any prior knowledge of the target system. However, the approach is very difficult to determine threshold which an anomaly should be considered intrusive.

Statistical method also used to analyzes system calls in privileged processes with discriminant analysis with Mahalanobis distance (Midori *et al*., 2001). This method appears to be quite efficient utilizing only eleven system calls to discriminate between normal and intrusive activities. Moreover, the approach does not require user profiles or a user activity database in order to detect intrusions. However, their method has a problem; the result of analysis for all samples in their method still have high misclassification in order to discriminate between intrusion and normal activities.