PARALLEL NETWORK ALERT MANAGEMENT SYSTEM FOR IDS FALSE POSITIVE REDUCTION

HOMAM REDA KAMEL EL-TAJ

UNIVERSITI SAINS MALAYSIA

2011

ACKNOWLEDGMENTS

"All praises and gratitude to ALLAH almighty"

This modest research will never become true without the contributions from many special relatives, friends and colleagues in their own different ways. For that reason, I would like to extend my appreciation to the following.

First of all, the praises and thanks go to almighty ALLAH for giving me the patience, the guidance, the health as well as giving me the chance of working in such an environment in UNIVERSITI SAINS MALAYSIA (USM) and in National Advanced IPv6 Center of Excellence (NAv6) particularly. Second, I deeply thank my supervisor Dr. Omar Amer Abouabdallah the one who taught me the meaning of help and support. A special thanks to him for guiding me all through the way to achieve this research and giving me the chance to contribute to this field. I also would like to thank my co-supervisors Dr. Ahmad Manasrah and Dr. Chan Huah Yang.

Special thanks to USM Fellowship Scheme respectively, and to NAv6 colleagues for providing a conductive environment and support during research.

Last but not the least, I would like to thank those who are close to my heart; my respectful *Father* Dr. Reda El-Taj (My Example), for his endless, support, and continuous encouragement to follow his path in the academic field, to my beloved *Mother* Elham Mubaslat for her care, love and praises, to my dearest *Brother* Essam El-

Taj for his making my dream become true, to my precious *Sister* Hamah El-Taj for teaching me how to keep my smile, to Dr. Adnan Hnaif and Dr. Osama Alia, to all my close friends in Malaysia Dr. Mahmoud Haddad, Muhannad Abu Hashem and Mohammed Al-Momani, to all my colleagues in NAv6 the Brothers Band Dr. Manzoorsk Kolhar, Moein Mayeh, Ahamed Al-Madi, Mohammed Al-Halabi and Mohammed Anbar. Finally I would like to express my thanks to the Unknown Soldier who supports me without notice. I dedicate this work to all of them as without their support and understanding, this thesis would not have been completed.

Thank you!

Homam Reda El-Taj

Penang, Malaysia. June 2011.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ACC	aggregation and correlation components
AgC	aggregation component
App-IDS	Application intrusion detection systems
FCM	fuzzy cognitive maps
GCA	Group Correlation Algorithm
GCA	Group Correlation Algorithm
GCF	Group Correlation Framework
HIDS	Host intrusion detection systems
IDMEF	Intrusion Detection Message Exchange Format
IDS	Intrusion Detection Systems
IDSAMS	IDS Alerts Management System
IPS	Intrusion Prevention system
MIMD	Multiple Instruction, Multiple Data
MISD	Multiple Instruction, Single Data
NIDS	Network intrusion detection systems
P-GC	Parallel Group Correlation
P-GCA	Parallel Group Correlation Algorithm
SIMD	Single Instruction, Multiple Data
SISD	Single Instruction, Single Data
SPADE	Statistical Packet Anomaly Detection Engine
TAF	Threshold Aggregation Framework
TIAA	Toolkit for Intrusion Alert Analysis

TTA	Time Threshold Aggregation algorithm

LIST OF APPENDICES

SISTEM PENGURUSAN KESELANAN TERAGIH UNTUK MENGURANGKAN IDS POSITIF-PALSU

ABSTRAK

Setiap sistem keselamatan mempunyai kemungkinan untuk gagal. Justeru, usaha yang lebih perlu diambil untuk melindungi sistem-sistem ini. Sistem Pengesanan Pencerobohan (IDSs) telah dicadangkan sebagai perlindungan tambahan bagi sistem-sistem keselamatan. IDS merupakan suatu sistem keselamatan digunakan untuk melindungi persekitaran komputer. Ia mencetuskan ribuan amaran sehari yang membolehkan juruanalisis keselamatan mengesahkan kerelevanan dan keterukan setiap amaran berdasarkan kepada kriteria pengagregatan dan korelasi. Beberapa kaedah pengagregatan dan korelasi telah dicadangkan untuk mengumpul amaran tersebut. Tesis ini membentangkan satu Sistem Pengurusan Amaran IDS (IDSAMS) baru yang merupakan suatu sistem selari untuk mengurus amaran IDS, mengurangkan positif palsu melalui pengagregatan dan korelasi amaran IDS bagi memberi kefahaman sepenuhnya tentang serangan pada rangkaian di samping memudahkan kerja juruanalisis serta menjimatkan masa mereka.

IDSAMS merupakan satu sistem kendiri yang berfungsi berdasarkan amaran nyata daripada data dalam atau luar talian menjadi Sistem Penyiasatan Forensik. Sistem ini dibangunkan daripada gabungan algoritma pengagregatan dan korelasi. Setiap satu daripada dua algoritma tersebut telah dilaksanakan untuk membentuk satu sistem kendiri yang lengkap. Sistem pengagregatan bertujuan menyingkirkan lebihan daripada fail amaran dan mengurangkan positif palsu. Sementara sistem korelasi bertujuan untuk menyingkirkan positif palsu, mengurangkan jumlah amaran dan menyelesaikan perkaitan antara amaran. IDSAMS bertujuan membantu juruanalisis memperoleh gambaran keseluruhan yang kukuh tentang aktiviti berkaitan keselamatan pada rangkaian.

IDSAMS menggunakan algoritma korelasi yang dikenali sebagai Korelasi Kumpulan Selari (P-GCA) yang meningkatkan Algoritma Korelasi Kumpulan (GCA). GCA menghubungkaitkan keputusan pengagregatan tersaring untuk memberikan keputusan yang lebih baik dan tepat dalam masa yang singkat. GCA dibangunkan daripada algoritma pengagregatan yang dikenali sebagai Algoritma Pengagregatan Ambang Masa (TTA). Penggunaan keselarian dalam sistem pengagregatan dan korelasi mampu meningkatkan kelajuan bagi mendapatkan keputusan akhir daripada amaran yang terkorelasi.

Keputusan menunjukkan bahawa IDSAMS menghubungkaitkan amaran IDS secara tepat berdasarkan permintaan pengguna dan nilai ambang. Keputusan IDSMAS memberikan dua tahap yang berbeza: Tahap pertama adalah sistem pengagregatan untuk menyingkirkan amaran yang berlebihan. Sistem pengagregatan mengurangkan jumlah amaran sebanyak 41% jika pilihan adalah 8 sifat dan 50% dengan 4 sifat pilihan. Keputusan ini dianggap sebagai tahap pertama untuk mengurangkan positif palsu daripada fail amaran heterogen IDS. Tahap kedua adalah keputusan sistem korelasi dengan peningkatan 94 peratus daripada keputusan pengagregatan. Oleh itu, sistem korelasi berfungsi ke atas keputusan pengagregatan (41 peratus dan 50 peratus) yang memberikan 6 peratus daripada fail amaran heterogen asal. Peningkatan keseluruhan adalah 97.84 peratus daripada amaran sebenar tanpa amaran positif palsu dan amaran berlebihan. Keputusan selari menunjukkan bahawa sistem pengagregatan dan korelasi dapat mengurangkan masa pemprosesan keseluruhan dengan kelajuan adalah dalam julat 1.406 sehingga 1.614 dengan teras dual dan 2.14 sehingga 3.23 dengan teras kuad.

PARALLEL NETWORK MANAGEMENT SYSTEM TO REDUCE IDS FALSE POSITIVE

ABSTRACT

Every secure system has the possibility to fail. Therefore, extra effort should be taken to protect these systems. Intrusion detection systems (IDSs) had been proposed with the aim of providing extra protection to security systems. IDS is a powerful computer security system used to secure the computer environments. These systems trigger thousands of alerts per day, which prompt security analysts to verify each alert for relevance and severity based on an aggregation and correlation criterion. Several aggregation and correlation methods have been proposed to collect these alerts. This thesis presents a new IDS Alerts Management System (IDSAMS) which is a parallel system used to manage the IDS alerts, reduce the false positive by aggregating and correlating the IDS alerts to give full understanding of the network attacks as well as easing the process for the analysts and save their time.

IDS Alerts management system is a standalone system which can work based on real alerts from an online data or offline data as full a Forensic Investigation System. This system was built by combining the aggregation algorithm and correlation algorithm. Each one of these two algorithms had been implemented to form a complete standalone system. The aggregation system aims to remove the redundancy from the alert's file and reduce the false positive, while the correlation system aims to remove the false positive, reduce the alerts amount and resolve the relations between the alerts. IDSAMS aims to help the analyst to have a strong overview of security-related activities on the network.

IDSAMS employs a correlation algorithm called Parallel Group Correlation (P-GCA) which is an enhancement of our Group Correlation Algorithm (GCA). GCA correlates the filtered aggregation results to give better and accurate results in a short time. GCA was built over an aggregation algorithm called Time Threshold Aggregation algorithm (TTA). The use of parallel in the aggregation system and the correlation system is to enhance the speed up of getting the final results of correlated alerts.

Results show that IDSAMS correlates IDS alerts accurately based on user demands and threshold value. The results of IDSMAS have two different stages: First stage is the aggregation system to remove the redundant alerts. The aggregation system reduced the amount of alerts by 41% if the choice is 8 features and 50% with 4 features choice. These results consider as the first step of reducing the false positive from IDS heterogeneous alerts file. The second stage is the correlation system results where the enhancement is 94% of the aggregation results. The correlation system works on the aggregation results (the 41% and the 50%) which give us 6% of original heterogeneous alerts' file. The overall enhancement is 97.84% of real alerts without false positive and redundant alerts. The parallel results show that the overall processing time had been reduced by parallelizing the aggregation system and the correlation system, the speedup results was on the range of 1.406 to 1.614 over Dual cores and 2.41 to 3.23 over Quad cores.

CHAPTER ONE

INTRODUCTION

In the recent past, networks and their applications have become indispensable in the creation of any organization. Local area networks (LAN) have become more complicated in their infrastructure, since each organization network requires specific infrastructure, and thus numerous sub-networks to fulfill the needs of each department. On top of that, wireless networks can be added to these organizations to ease the connectivity process for the users through their personal laptops or smart phones. All of these create complex, highly dynamic networks. Above all, security in the Internet has become one of the urgent topics that should be addressed. Security has become a primary concern because of these complicated issues.

1.1 Security Issues

In the past, computers were not connected to each other owing to the absence of a network to connect them. Subsequently, the concept of network with the capacity to connect computers in the same companies, in the same building was established. Nowadays, approximately all computers are connected to the Internet. The main concern with this issue is the increased number of possible attackers that can harm any system. Moreover, an attacker can assail any computer either by connecting to the same local area network or from almost anywhere in the world.

Storing important and sensitive data in computers has made the issue of security more important than ever. Previously, students' or employees' records were stored by filing them as hardcopies in an archive. Keeping records safe used to be an easy process. Stealing or copying files requires a physical attack. At present, records are stored in computers, making them more vulnerable to spiteful attackers. Storing records in computers makes the attacking process easier. All that the attacker needs is a remote connection from an outside location, without necessarily entering the location of the stored records. Storing records and data on paper remains important, like in the case of banks, where the customer's account transactions are recorded in receipts that serve as evidence of these transactions, if the system collapsed. This is important because financial issues are very sensitive.

At present, there exists Internet banking systems, which do not provide the customer with an official receipt bearing an official stamp. Moreover, other companies, such as telecommunications firms, give discounts to customers who are amenable to non-paper-based receipts when using either the e-payment or the normal payment method.

Most businesses rely on computer systems to execute their data work. This reliance has become full dependence, such as in the case of Internet-based businesses, where a lack of connection or network failure will result in no business. Furthermore, an unplanned computer shut down may cause a leak in data.

Some organizations rely on paper work to support their Internet-based systems. They calculate the price of a chosen item, print out the receipt, and send it by regular mail.

The changes in the way that businesses deal with storing and retrieving data increase the number of potential attacks, and consequently, successful attacks become more serious.

Successful attacks also enable attackers to improve their techniques. At present however, it is rare to see a single successful attack on a single computer because it takes multiple attacks for one to be successful. Moreover, these attacks can be on one or more computers.

These kinds of attacks have become a challenging issue, and it is hard to protect a network against them. Furthermore, it is not easy to know if the attackers used multiple attack techniques, especially if their attacks originate from outside the network.

Computer security is an important topic. Securing and limiting access to private and sensitive data and ensuring the availability of critical security systems are important. Computer security should accomplish these goals.

1.2 Security Techniques

Security in computer systems can be achieved using three techniques: *authentication, authorization, and accounting.* These techniques are necessary to protect the computer network against brutal attacks (Etoh & Wiley, 2005; Niemi, 2002; Todoro, 2007).

Authentication is the process of giving personal information to the computer system. The authentication process requires a saved user name and a password to match it. The assumption here is that only the owner of that personal information with its matching password will be able to log on to the system. This will enable the system to know the identity of the users who log on to it or to retrieve the information from the system memory if needed.

There are various ways to do the authentication process, either by using one process or by combining two or more processes. Examples of these processes are: typing in information that the user knows, such as the password; using a smart card or special device; and using a part of the user's body like the eye, thumb, or voice.

Authorization is the process of checking if the operation is being performed by the authorized logged on user. Each system grants privileges; if a user does not have the privilege to perform an operation, the access would be blocked. Authorization will rely on the authentication process in order to function. If authorization fails, a user who has the intention of gaining access to secured data can cheat the system. The authorization process is performed by the system before it gives file access permission to a user, and this process will be executed each time a user tries to access any protected resource when logged on, based on the user's privileges.

Accounting or auditing is the process of recording security data related to a user's activities. The data will be stored in a log file, which may contain the user ID, time of login, time of logout, the IP used for login, and any other information related to the user resource files he or she tried to access. The idea behind auditing is to keep a record of any type of authorized or unauthorized access to any resource.

Another related topic is cryptography, which is not a security technique in itself. However, this is very important to be able to secure the sensitive traffic data and to prevent outsiders form hacking into the data. However, data encryption and decryption during sending and receiving sensitive data can be easily broken if there is any leak between the server and the login user computer (Stallings, 2011).

1.3 Security Techniques Failures

When there is permeation of the security, all the sensitive data in the system will be at real risk. Permeation can be done if a fault exists in the security. Techniques such as *Bypass* security techniques, *Spoof* security techniques, and *Guess* security techniques can be implemented (Löf et al., 2010).

Bypass security technique is designed to enable the attacker to access the protected data without interrupting the system and allows the attacker to listen to the buffer overflow. As the user enters his personal data such as the password in a specific part of a memory, the attacker will implement his or her code in such way that he or she is enabled to recall the address of that function which is responsible for writing on a specific part of memory. Consequently, the attacker will be able to run any code that he or she demands. Thus, any authorization or access controls can be bypassed easily (Thompson, 2003).

Spoof security technique is designed to facilitate changing the authenticated identity of the attacker. For example, when the technique is successful in attacking the authentication based on IP, it becomes easy for the attacker to spoof the IP and use it to change his identity.

Guess security techniques is based on guessing. Thus, if the password is guessable, the attacker can run a dictionary attack and login as a system member-user. Other techniques are user-dependent. When editing the features of a system configuration, unwanted programs can be installed. In other instances, the end user will be tricked into revealing information. This technique is called *social engineering* attack. These types of attacks are possible, even in the most highly secured systems.

1.4 Intrusion Detection System

Since any secure system can fail, extra effort should be provided to protect these systems. One such protection method, the intrusion detection systems (IDSs) has been proposed. Generally, intrusion detection has different meanings according to context. For instance, intrusion is defined as an action that successfully abuses certain security procedure (J. Zhou, Heckman, & Reynolds, 2007). IDS monitors the whole system in order to trigger alerts in the presence of any intrusion or malicious behavior. The system administrator will deal with the report of alerts to choose the proper action for each alert (Bace & Mell, 2001; Horng et al., 2010; Lauf, Peters, & Robinson, 2009).

1.5 Intrusion Detection Systems Challenges

IDS has become a required tool in any corporate network architecture. However, the way of detecting intrusions has remained imperfect. The main problem with IDSs is that they generate a huge amount of alerts that are not caused by real attacks (Tjhai, Papadaki, Furnell, & Clarke, 2008). These false alerts are referred to by the term false positive. False positive is a mystery term that describes the situation wherein the IDS triggers an alert when a malicious activity is initiated even in cases where the activity is not malicious. In other words, IDS can make a mistake (Pinyathinun & Sathitwiriyawong, 2009). Nevertheless, the IDS becomes a forensic tool since most administrators look into the alert log file to make their investigations after the intrusion has occurred.

Another problem with the present IDSs is the irrelevant alerts that the IDS triggers (e.g., a warning alert about a web-based attack that only works against Windows computers, while the target is a Linux computer). IDS alerts do not have sufficient information for the administrators to base their decisions on with regard to the right action to be taken for each intrusion. Any network environment contains different types of IDS sensors (NIDS, HIDS, and App-IDS) and each type is concerned about specific types of data. While the NIDS alert will cover the IPs and Ports, the HIDS will cover the buffer overflow. Given that all sensors will save their alerts in the same heterogeneous log file, dealing with these different structures of alerts will be an annoyance for the administrators. Moreover, since there are many sensors involved in monitoring the network and all sensors pump in their alerts in the same heterogeneous log file, the huge problem created by redundant alerts becomes an issue. The redundant alerts are sometimes the same alerts but with different sensor IDs, or possess a slight difference in the time stamp. Further explanation is provided in chapters two and three.

1.6 Intrusion Detection Aggregation Systems

Aggregation as a technique is a major part of correlation techniques which can reduce the complexity involved in alert analysis (P. Ning, Cui, & Reeves, 2002a) because the correlation will execute the alerts based on a specific classification. This step will be done after the aggregation has removed the redundant alerts to reduce their volume (H. Debar & Wespi, 2001). Removing the redundancy will depend on the similarity of the alerts' features.

1.7 Problems with the Existing Aggregation Systems

The main problem with all of the proposed alert aggregation systems is that they deal with it as a part of the correlation systems and they do not provide a complete aggregation solution. Instead, they only concentrate on a limited part of the aggregation process since one alert can represent all redundant alerts. For instance keeping redundant alerts because of the difference on sensor IDs will delay the process of correlation. The same problem will appear if there is a slightly different in time stamp between the alerts. The main problem with all the proposed alert aggregation systems is the manner with which they deal with aggregation as part of the correlation systems and thus, they do not provide a complete aggregation solution. Instead, they only concentrate on a limited part of the aggregation process since one alert can represent all redundant alerts. For instance, keeping redundant alerts because of the difference on sensor IDs will delay the process of correlation. The same problem will appear if there is a slight difference in the time stamp between the alerts. Failure in removing the redundant alerts will delay the process of correlation that may lead to inability of preventing a successful attack because action from the analyst has not yet been made.

1.8 Intrusion Detection Correlation Systems

In order to improve some of the intrusion detection systems problems, the alert correlation has been proposed. Alert correlation techniques deal with the IDS alerts after these have been collected in one file as heterogeneous file of multiple sensors. The purpose of correlating the alerts is to create high-level reports of the network status. The main goal of correlation is to achieve the minimum amount of alerts by reducing the false positive alerts and the redundant alerts by aggregation methods as well as ranking and grouping the alerts that refer to the same event. The correlation techniques will classify the alerts into different classes based on their features such as IP addresses and port numbers. Consequently, the higher ranking of overall alerts with same feature similarity in the same class will be correlated (P. Ning, Y. Cui, D. S. Reeves, & D. Xu, 2004). Alert correlation has been studied from different perspectives such as *Attack scenario, Multi-stage,* and *Filtering.* Attack scenario algorithms have been proposed because of the idea that normal attacks most probably will be executed in single attacks while complex attacks are executed in several episodes (Dain & Cunningham, 2002). Multi-stage address is the problem of detecting unknown attack. This approach is mainly based on the assumption that the attacks perform multiple steps to fulfill the global intrusion (Cuppens & Miege, 2002).

Filtering-based approaches have been proposed to remove the need for complicated attack library and to reduce the amount of unrelated or irrelevant alerts. These approaches are based on non-existing services. Filtering-based approaches consider the topology and the operational objectives of the protected network during the alert correlation (P. Porras, M. Fong, & A. Valdes, 2002).

1.9 Problems with the Existing Correlation Methods

The proposed correlation methods only cover a limited part of the correlation process. Thus, several problems have not been addressed. For instance, multi-step correlation executes the alerts in three steps, namely, managing, clustering, and merging. This technique does not reduce the problem of the high false positive rate of IDS alerts. The quality of the heterogeneous alert file affects the correlation process and it will consume more time to be correlated (Liu, Zheng, & Yang, 2010).

1.10 Problem Statement

The current correlation methods deal with one type of alerts but cannot handle the combination of multiple alert types. Existing methods cannot be easily extended, and they only operate in online or offline data.

This thesis presents an online-offline IDS alert management system which can also be used as a forensic tool. The system supports any type of alerts with minimum requirements. It can be easily updated by adding new aggregation or correlation components. The system requires minimal processing to produce the final reports after implementing the two processes of aggregation and correlation. The components (subsystems), aggregation, and correlation can work together or independently. The performance of each subsystem and the overall performance of the framework were evaluated using different datasets from multiple IDS audit data.

1.11 Objectives of the Study

The objectives of this thesis are:

- To reduce the amount of false positive alerts, noise (rubbish) and non relative alerts from IDS alerts.

- To enhance the forensic studies by giving the choice to the user to group the IDS alerts either by the aggregation or the correlation based on online data or offline data.
- To use the Multi-Core technology OpenMP in aggregating and correlating the alerts.

1.12 Contributions of this thesis

IDS alerts analysts suffer from analyzing IDS alerts because of the huge amount of redundancy and false positive (Zurutuza & Uribeetxeberria, 2004). This thesis provides solutions for the problems outlined above, and provides the following contributions:

- IDS alert aggregation framework: New aggregation framework called Threshold Aggregation Framework (TAF) to aggregate IDS alerts based on time threshold, this framework proposed to enhance the IDS alerts filtering and to remove the redundant alerts.
- Filtering and parsing method: this method aims to filter and parse the incomplete alerts (alerts with missing features) by modifying them to reduce the processing time of aggregation the IDS alerts.
- IDS alert correlation framework: New correlation framework called Group Correlation Framework (GCF), this framework based on aggregation filtered results, proposed to smooth the alerts to give a better relational alerts, with less amount of original alerts and reducing the false positive alerts.

- New correlation algorithm: this algorithm called Parallel Group Correlation (P-GC), this algorithm is introduced to enhance the correlation speed based on Multi-Core technology (OpenMP).
- False positive reduction by two steps after the aggregation by removing the redundant alerts and after the correlation by removing none correlated alerts.
- IDS Alerts management system: This system is a standalone system which can work on online alerts or offline alerts to be used as Forensic Investigation System, the aggregation and correlation frameworks together form a complete system aims to help the analyst having a condensed overview of security-related activities on the network.

1.13 Research Methodology

The methodology of our research has two parts: Aggregation and Correlation.

- IDS Alerts Aggregation; the methodology of aggregation based on the Threshold Aggregation Framework (TAF), this framework works on the similarity between the alerts' features based on a threshold time value (the concept based on mathematical *Row Echelon Form*). TAF has three main components: the *Data Controller*, the *Framework Core* and the *Aggregation Controller*.
- IDS Alerts Correlation; the methodology of the correlation based on Group Correlation Framework (GCF), this framework works on *Apriori* algorithm (mathematical data set combination) to find the relationships between the alerts based on the grouped alerts from the aggregation framework TAF (the output of TAF is the input of GCF).

1.14 Thesis Organization

The remainder of this thesis is structured as follows. Chapter two will give an over view of the related works in the area of reducing the false positive of IDS alerts, aggregating the IDS alerts and correlating them, at the end of this chapter we give a comparison between the related works and our research work. Chapter three will give a closer view on our methodologies of aggregating and correlating the IDS alerts with more explanations on the IDS Alerts Management System (IDSAMS) and its two subsystems. Chapter four describes the algorithms of the IDSAMS subsystems, the mathematical proof of the algorithms and gives an implementation details on both of the two subsystems. Chapter five illustrates the evaluation of the components and subcomponents of each subsystem of the IDSAMS, explains the results we achieved and compare our results with the existing related system. Chapter six highlights the conclusion of the whole research and gives a forward view of the future work.

CHAPTER TWO

LITERATURE REVIEW

This chapter investigates the related works on IDS false positive problem. Previous works on IDS false positive employed two different perspectives based on the reduction process. The false positive reduction process can be done either at the sensor level as a part of intrusion detection systems or after the intrusion detection process on the log alert file as a forensic method.

In this chapter, we discuss the false positive problem and the ways of reducing the amount of the false alerts in the IDS alerts log file by aggregating and correlating the alerts.

2.1 Classification of IDSs

IDSs can be classified in numerous ways. The general way of classifying is based on detection method, audit source, usage frequency, and behavior of detection method (H Debar, Dacier, & Wespi, 1999).

2.1.1 **IDS** Based **on Detection Method**

IDS has two main intrusion detection techniques. These are *anomaly detection* and *misuse detection*. Anomaly detection technique determines the abnormality by measuring the distance between the suspicious activities and the norm based on the chosen threshold. Misuse detection technique searches for malicious signature or pattern based on the set of rules or signatures to detect intrusive behavior. Difference between these two IDSs exists. The misuse detection cannot detect novel attacks but it has a lower rate of false positive alerts while anomaly detection detects the novel attacks with a higher rate of false positive alert. However, combining these two detection approaches will provide a more efficient detection method with a lower rate of false positive alerts (Alharby & Imai, 2005).

Misuse detection methods are based on two types either stateless or stateful systems. Stateless systems examine each event separately in the input stream to determine whether an event is malicious or not; stateful systems examine the events and the relationships between them to know the history of events in each attack. Stateful systems can support more complex rules than stateless systems which make these systems more expensive in relation to CPU usage and memory space (Vigna, Robertson, Kher, & Kemmerer, 2003).

Anomaly detection methods fall into two categories: *learning-based* or *specification-based*. Learning-based anomaly detection method is based on neural networks and consists of two phases, namely, a training phase and detection phase. The training phase aims to teach the system about the properties of the normal traffic by assuming that the training audit data contain zero attacks; otherwise, the system will include these attacks in the normality traffic. After the training phase is conducted, the system goes to the detection phase. In this phase, the input traffic data are compared with the trained traffic data from the first phase. The system will report an anomaly and trigger an alert if there is any audit event that does not match the learned traffic data (Das, Matthews, Srivastava, & Oza, 2010).

Specification-based anomaly detection methods depend on a specification of the normal traffic type. The specification can be specified as either manually or automatically generated. Automatic specification relies on the user manual specification and depends on human experience to define the properties of events. After defining the specification, the system will start detecting the intrusions by comparing all legal sequences generated by a program with the current sequences of the program (Najjar & Azgomi, 2010; Wagner & Dean, 2001).

2.1.2 IDS Based on Audit Source

Another way of classifying IDS is by the audit data that are used to detect the intrusions. There are three categories of audit data: *Network intrusion detection systems* (*NIDS*), *Host intrusion detection systems* (*HIDS*), and *Application intrusion detection systems* (*App-IDS*) (Najjar & Azgomi, 2010).

NIDS deal with the packets from the protected network. Detection in these systems deals with the full packets, packet headers, or Netflow data. Other NIDS deal with the firewall logs to check the packet headers of the blocked packets (Corchado & Herrero, 2010; Sekar, Guang, Verma, & Shanbhag, 1999).

HIDS are the first invented intrusion detection systems. The target environment for the detection is a mainframe computer since the interaction from outside is rare. The intrusion detection analyzes the audit information provided by the mainframe, either locally or by a separate machine, and reported as suspicious security events (H Debar, et al., 1999; Najjar & Azgomi, 2010).

App-IDS deal with the log which has been created by a user space application. Application intrusion detection systems solve some of the problems that might be present in the network-based detection. This approach has the ability to perform intrusion detection analysis at different stages based on client requests (Vigna, et al., 2003).

2.1.3 IDS Based on Usage Frequency

IDS based on user frequency IDS can be classified as either *Online Systems* which deals with real audit data at the time the data are generated, or *Offline Systems* where the system is run frequently to look for signs of attack based on a period of time.

2.1.4 IDS Based on Behavior of Detection

This classification is based on the type of response when an intrusion occurs. The response either be *passive* based, meaning where the administrator will be notified after an intrusion happened by such means as email, or *active* based on the logic of blocking the attacks as a prevention. This system is called *intrusion Prevention system* (IPS) (Ierace, Urrutia, & Bassett, 2005).

2.2 False Positive Problem

False positive problem is a mystery term that describes the situation where the IDS triggers alerts when a malicious activity occurs even if the activity is not malicious. Thus, IDS makes a mistake (Ranum, 2003). IDS triggers thousands of alerts, 99% of which are false positive (Furnell et al., 2008; Jamdagni, Tan, Nanda, He, & Liu;

Spathoulas & Katsikas, 2010; Yusof, Selamat, & Sahib, 2008; Zurutuza & Uribeetxeberria, 2004).

Organizing and dealing with the recorded logs and generated alerts by the security sensors such as the IDS, firewalls, packet filtering, and servers are not easy. Most of the organizations consider these alerts as a major problem. Since these sensors are independent, they will trigger alerts and send these to the analyst for analysis towards understanding the nature of the intrusion using the provided tools, methods, and techniques. This leads to the reduction of the false alert rate and the increase of the attack detection rate. In spite of these, weaknesses still exist in these processes because of the quality of the input data. Dealing with huge number of alerts containing equally large number of false alerts will be the manner with which any sensor works even when a harmless event occurs (Yurcik, 2002; Yusof, et al., 2008). False positive can be reduced during the time of detection of the attack (sensor level) or after the detection (on the heterogeneous alerts log file).

2.2.1 Sensor Level False Alerts Reduction

Reducing false positive alerts during the detection of the attacks was addressed by many studies using different techniques and methods like the fuzzy cognitive maps (FCM) which is a soft computing modeling techniques generated from the compensation of fuzzy logic and neural network. In this proposed solution the measurements based on (availability, similarity, occurrence, relevancy, independent and correlation factors), the second step is to assign an effect value for each one of the factors to estimate the total degree of abnormality per packet. Depending on the factor value the packet will be dropped or ignored. That if the packet is below malicious and if not it will be considered as real alert in the (FCM). The last step is to measure the (effect/influence) value and there is a degree from 0 to 1, while 0 means normal relation and 1 means high relation. This study shows that improving the detection deficiency will be by reducing the false alerts and increasing the detection accuracy at the sensor level (Jazzar & Bin Jantan, 2008).

Lui used agents and data mining technology to give more accuracy by capturing the actual behavior of network traffic. There are three types of agents for the three data mining techniques: (clustering, association rules and sequential association rules). The number of agents will be different in both training and detection process, the clustering-base agents extracts properties from traffic in terms of frames and tries to make the normal traffic in the training stage. If the unknown traffic is far from the normal cluster it is classified as an attack. The association rule-based agent finds out the relationship between features selected and traffic property in the training phase. The agents will capture the rule of selected features and in the detection phase, the agents count the rules of each connection to be matched, when the frequency is less than the threshold it classified as an attack. The sequential association rule-based agents (in the training phase) capture the sequential patterns in network traffic dialog to assist the association mining process. In the detection phase the agents tests the abnormal connections matched within the (packet/time) frame. If it is larger than the threshold, it will be declared as an attack. In the decision maker stage, they check if the alert is generated from both clustering based and rule based to declare an attack, else it will be a false alert from one side which will be eliminated by the other side (Lui, Fu, & Cheung, 2005).

Pi-Cheng made an optimization of the rule selection and the attack identification in attack analysis, by proposing a scenario-based approach to correlate malicious packets and to select intrusion-detection rules in intelligent way. The scenario-based approach is based on how to choose rules to be tested according to the threats detected and attack scenarios identified at the moment of the attack. Instead of being tested simply according to some predetermined order, depending on a dependency graph which is a direct acyclic graph, the main idea of this approach is to classify rules in the rule database in terms of threats and thus associate the rules with a dependency graph (Pi-Cheng, Chin-Fu, Tei-Wei, & Juan, 2005).

2.2.2 False Alerts Reduction after the Detection Level

Reducing the false positive after detection means reducing the amount of false alerts in the alert's file. In other words, the false alerts are differentiated from the real alerts. In order to know the differences between real and false alerts, researchers began studying and analyzing the alerts' features. This resulted in the formulation of three approaches: alerts aggregation, alerts correlation, and the combination of both.

Alerts Aggregation Techniques

Aggregation techniques have been used in many fields in computer science to minimize the analysis time and to reduce redundant packet transmissions in networks such as the centralized, tree-based, static-cluster, and dynamic-cluster aggregation schemes (Park, 2006). IDS aggregation technique studies are used for grouping and minimizing the alerts to facilitate the process of analysis and removal of redundant alerts. Aggregation techniques are based on the similarity of features because several alerts are related to a similar event. Generally, these alerts have similar features and many sensors can trigger the same alert on similar attacks with different formats. Thus, numerous alerts are triggered which should be aggregated into one alert.

Kanamaru proposed a packet aggregation technique to aggregate the packets in order to minimize analysis time of the fault packets to be detected. The proposed aggregation model character is based on sender and receiver addresses (Kanamaru *et al*, 2000). Depending on these addresses, three combination models have been formulated:

1) ***1 – 1 Model*** This model concern on the flow from a specific source address to a specific destination address.
2) ***1 – M Model*** This model concern on the flow from a specific source address to any destination address.
3) ***M - 1 Model*** This model concern on the flow from any source address to a specific destination address.

The aggregated packets from the whole captured packets were computed by using the following formula:

$$\frac{aggregation\ packets}{all\ captured\ packets} \times 100 = aggregation\ rate(\%) \qquad (2.1)$$

This study shows that the result of using the combination model is effective in characterizing packets. The administrator can obtain useful information for fault detection from the aggregated packets. In addition, the administrator can decide if the problem is of low severity (between two hosts) or high severity (between router and server). This model has been used to detect fault packets but never as a model to detect false alerts. The evaluation was based on capturing real (385786) packets from the network to obtain (161778) aggregated packets of 191 symptoms. The aggregated percentage was 42% of the original capture packets.

Valdes proposed a framework for a general aggregation algorithm by including five features of the alerts' features. These are *source IP addresses, source ports, destination IP addresses, destination ports, and alert generation time* (Valdes & Skinner, 2001). The compression result of each feature is a value between 0 and 1, while the similarity calculation and the weights of each feature depend on configuration definition. Aggregation of the alerts will be done depending on the predefined weights and the compared result of each feature. The overall similarity between any two alerts is computed by the following equation:

$$SIM(X,Y) = \frac{\sum_j SIM(X_j, Y_j)}{\sum_j E_j} \qquad (2.2)$$

Where *X* = Candidate meta alert for matching

Y = New alert

j = Index over the alert features

Ej = Expectation of similarity for feature j

X j,Yj = Values for feature *j* in alerts *X* and *Y*, respectively.

This approach has been evaluated by experimenting on 4439 alerts triggered by IDS sensor to obtain 604 Meta alerts. The result was 14% Meta alerts of the origin alerts file. Another proposed solution was based on the exact matching of three features (*IP source, IP destination, and Alert class*) (H. Debar & Wespi, 2001). The aggregation of the alerts was performed by seven levels based on the similarity of Source IP (S), Destination IP (D), and Alert Class (C). The seven levels are:

{*Same (S, D, C), Same (S, D), Same (S, C), Same (D, C), Same (S), Same (D) and Same (C)*}. A threshold value for choosing the level of alerts aggregation is involved. The aggregation reduces slight amount of alerts because the aggregation process starts after the alerts have been correlated.

A different result was achieved from the study of (Valdes & Skinner, 2001) by (Mu, Huang, Tian, Lin, & Qin, 2005) The authors proposed a method that uses fuzzy matrix in the final evaluation to achieve a slightly better experiment results with the use of only the source IP addresses and alert generation time.

Autrel proposed another aggregation technique to categorize alert features into four classes based on the Intrusion Detection Message Exchange Format (IDMEF) node class which contains *Alert Location, Alert Name, and Alert Address* (Curry, Debar, & Feinstein, 2002) (Autrel & Cuppens, 2005). The proposed aggregation technique, which was based on the similarity operator, was used to compute the similarity of the same class of features in order to obtain a similarity value between two alerts. Calculation of the similarity values between attributes of the alerts was done which allowed aggregation. Consequently, the final similarity value was obtained. In this aggregation technique, no discussion on the weight function for each class was conducted. The evaluation performed by developing a system to aggregate (3776) alerts in 64 clusters for two sets of 32 clusters for each cluster obtained a different target. Thus, they used the first 32 clusters with one target. The results using this approach achieved 1.7% aggregated alerts.

Dain suggested that the alerts should be first categorized on the basis of the attack scenario. For each existing scenario they calculated the probability that the new alert belongs to this scenario or not (Dain & Cunningham, 2002). The new alert was assigned to the scenario that produced the highest probability score. When all the scores were below the threshold, the new alert was not included to any scenario and instead, a new scenario was started. The alerts were aggregated by two approaches (*Naïve Technique and Heuristic Technique*). The naïve technique assumes that all attacks that belong to a single scenario share a common source address. The heuristic technique assumes that the new alert needs to be compared with the last scenario depending on the alert time. In the evaluation performed using three different amounts of datasets, the best results contained (246316) alerts. After aggregation, (4041) join decisions were obtained. The total result was 1.6% aggregated alerts.

Alerts Correlation Techniques

Correlation is part of the intrusion detection studies that smoothens the progress in the analysis of intrusion alerts based on the similarity between alert attributes. This can be represented mathematically as:

$$\mathrm{Corr}_{\mathrm{Alert}} = \{\mathrm{Alert}_1, \mathrm{Alert}_2, \dots, \mathrm{Alert}_\mathrm{n}\} \qquad (2.3)$$

The group of alerts $\{Alert_1, Alert_2, \dots, Alert_n\}$ with the same features and are related is represented by $Corr_{Alert}$. However, most of the correlation methods focus on IDS alerts by examining other intrusion evidence provided by system monitoring tools or scanning tools. The aim of correlation analysis is to detect relationships among alerts to promote building of attack scenarios (J. Zhou, et al., 2007; Zhu & Ghorbani, 2006; Zurutuza & Uribeetxeberria, 2004).

Classification of Alert Correlation Technique

IDS alerts correlation studies employ various perspectives and use many methods and techniques which can be categorized as similarity-based, pre-defined attack scenarios, pre-requisites and consequences, and statistical causal analysis (Xu & Ning; Yusof, et al., 2008; Zhu & Ghorbani, 2006).

i. Similarity-Based

Similarity-based approach attempts to estimate the similarities between two alerts depending on their features (Chen, Garcia, Gupta, Rahimi, & Cazzanti, 2009). IDSs trigger alerts when suspicious actions are monitored. These alerts have several features such as *source IP address, source port, destination IP address, destination port, alert classification, and timestamp*. Based on these feature values, similarity-based alert correlation approaches have been proposed to compute the similarities between two or more alerts, and subsequently group the related alerts together based on the extracted similar features.

These methods are closely related to data clustering techniques in data mining (Tan, Steinbach, & Kumar, 2006) because a group of similar alerts corresponds to the same attack or attack category. Similarity-based approach tries to reduce the number of alerts in the heterogeneous file that have been reported to the security analysts.