A STATISTICAL APPROACH TOWARDS WORM DETECTION USING CROSS-RELATION TECHNIQUE

by

MOHAMMED F.R. ANBAR

**Thesis submitted in fulfillment of the requirements
for the degree of
Doctor of Philosophy**

March 2013

ACKNOWLEDGEMENTS

بسم الله الرحمن الرحيم

{ نَرْفَعُ دَرَجَاتٍ مَنْ نَشَاءُ وَفَوْقَ كُلِّ ذِي عِلْمٍ عَلِيم "76" } (سورة يوسف)

The favor, above all, before all, and after all, is entirely Allah‟s (SWT), to whom my never-ending thanks and praise are humbly due.

I would like to take this opportunity to convey my sincere thanks and deepest gratitude to my supervisor, Prof. Sureswaran Ramadass for all the help and valuable guidance provided to me during throughout my period of research. I consider myself privileged to have had the opportunity to work under his guidance. I'm also grateful to my advisor Dr. Ahmed Manasrah and Mr Selvakumar Manickam for his help and support. Sincere thanks to all my friends especially, Dr. Ashraf Al Jammal and Eng. Alhamza munther for their help and support.

My acknowledgement also goes to all NAv6 centre members my colleagues, technicians, administrative staff, and NSST team. My Acknowledgement also goes to the Institute of Postgraduate Studies, and the university library for their help and support.

Moreover, I would like to thank those who are always in my heart; my father for his endless and continuous encouragement and constant support, my mother for her continuous prayers and inspiration. My sincere gratitude goes to my dearest brothers for their continuous supporting and encouragement. In addition to, my sisters for always keeping a smile on my face and motivating me all the time.

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATION

ACK	Acknowledgment
ANN	Artificial Neural Network
ASAWDCRT	Behavioral based Network worm detection and signature automation
DAW	Distributed Anti Network worm
DPCBWD	Destination port correlation based network worm detection
DSC	Destination source correlation
FN	False Negative
FP	False Positive
FPR	False Positive Rate
ICMP	Internet Control Message Protocol
IDS	Intrusion Detection System
IP	Internet Protocol
ITR	Improved Two Rotation
JPCAP	JAVA Packet Capture
NIC	Network Interface Card
NIDS	Network Intrusion Detection System
OS	Operating System
RST	Rest
SCANS	Statistical cross relation approach for network scanning
SYN	Synchronization
TCP	Transmission Control Protocol
TN	True Negative

TP	True Positive
TPR	True Positive Rate
UDP	User Datagram Protocol
WINPCAP	Windows Packet Capture
GTNetS	Georgia Tech Network Simulator
SBWD	Signature based network worm detection
LCS	longest common substring
DMZ	demilitarized zone
CPC	Classification Prediction Combined
CPS	Classification Prediction Separated

PENDEKATAN PERANGKAAN KE ARAH PENGESANAN CECACING MENGGUNAKAN TEKNIK HUBUNG SILANG

ABSTRAK

Rangkaian komputer telah menjadi satu dimensi penting bagi organisasi moden. Oleh itu, bagi memastikan rangkaian dijalankan pada prestasi tinggi (penggunaan rangkaian dan kelajuan biasa tanpa sebarang kesilapan) dianggap sebagai satu langkah penting bagi pertubuhan-pertubuhan ini. Untuk mencapai matlamat ini, rangkaian perlu selamat kerana keselamatan merupakan salah satu isu yang penting untuk mencapai tahap prestasi yang baik (tiada kesilapan dalam rangkaian seperti kegagalan sambungan pada kadar yang tinggi). Walau bagaimanapun, tugas ini adalah hampir mustahil terutamanya apabila terdapat isu-isu lain yang perlu ditangani. Tesis ini member penekanan pada pengesanan kehadiran rangkaian cecacing dalam rangkaian, yang merupakan salah satu masalah yang paling mencabar dalam keselamatan rangkaian. Dengan mengesan kehadiran rangkaian cecacing dalam rangkaian, sumber dan perkhidmatan boleh segera dilindungi dengan langkah-langkah keselamatan *patching* atau memasang, seperti *firewall*, sistem pengecaman pencerobohan atau sistem komputer alternatif.

Pendekatan yang sedia ada untuk pengesanan cecacing rangkaian dan pendekatan berasaskan perilaku terutamanya tidak cukup tepat untuk melakukan pengesanan cecacing rangkaian berdasarkan heuristik mudah dan pendekatan yang kurang berkesan untuk mengesan simptom rangkaian. Tujuan kajian tesis ini ialah untuk mencadangkan dan melaksanakan sebuah pendekatan yang mengesan kewujudan rangkaian cecacing dalam rangkaian.

Pendekatan yang dicadangkan dikenali sebagai Pendekatan Perangkaan ke Arah Pengecaman Cecacing Rangkaian menggunakan Teknik Hubung Silang (ASAWDCRT), terdiri daripada tiga sub algoritma iaitu: Rangkaian Pengimbas, Rangkaian Korelasi Cecacing dan Rangkaian Cecacing Tandatangan Automasi. Pendekatan rangkaian pengimbas terdiri daripada tiga modul yang kecil; (i) Modul Penapis (ii) Modul Penganalisis Trafik Statistik dan (iii) Modul Hubung Silang. Pendekatan Rangkaian Korelasi Cecacing, yang bertujuan untuk sumber destinasi mengesan korelasi perilaku untuk rangkaian cecacing dan terdiri daripada dua modul yang kecil; iaitu, (i) Modul pelabuhan destinasi korelasi berasaskan pengecaman cecacing rangkaian, (ii) Modul Amaran; dan (3) Pendekatan pengenalan rangkaian cecacing automasi tandatangan yang bertujuan untuk menjana pengenalan perilaku untuk gejala cecacing rangkaian (imbasan dan korelasi sumber destinasi).

Algoritma rangkaian imbasan adalah berdasarkan penggunaan paket TCP RST untuk mengesan imbasan berturutan TCP dan paket ICMP jenis ketiga (port yang tidak dapat dihubungi) untuk mengesan imbasan berturutan UDP. Imbasan rawak TCP disahkan apabila terdapat hubung silang antara ICMP jenis ketiga, kod 1 (port yang tidak dapat dihubungi) dan TCP RST. Imbasan rawak UDP disahkan apabila terdapat hubung silang antara ICMP jenis ketiga, kod 3 (port yang tidak dapat dihubungi) dan ICMP jenis ketiga, kod 1 (port yang tidak dapat dihubungi).

Dengan menggunakan ASAWDCRT, signifikasi keuntungan penindasan ialah 94.75% dari segi pengesanan ketepatan bagi pengesanan cecacing TCP dan 99.9% dari segi pengesanan ketepatan bagi pengesanan ketepatan UDP berbanding dengan skema pengesanan cecacing yang sedia ada. Sementara itu, algoritma imbasan diuji

dengan set data DARPA 1998 dan NAv6 dan disahkan algoritma imbasan adalah lebih cekap dan mengesan imbasan TCP dan UDP berbanding algoritma sedia ada dan juga menyediakan ketepatan pengesanan yang lebih baik dengan ketepatan purata 66.6%.

A STATISTICAL APPROACH TOWARDS WORM DETECTION USING CROSS-RELATION TECHNIQUE

ABSTRACT

Computer networks have become an important dimension of modern organizations. Thus, ensuring that networks run at peak performance (network utilization and speed running normal without any faults) is considered a crucial step for these organizations. To achieve this goal, networks must be secure because security is one of the essential issues for reaching a good performance level (no faults in the network such as high rate of connection failure). However, this task is next to impossible especially when there are other issues that need to be addressed. This thesis focuses on detecting the presence of network worms in network, which is one of the most challenging problems in network security. By detecting the presence of network worms in the network, resources and services can be further protected by patching or installing security measures, such as firewalls, intrusion detection systems, or alternative computer systems.

Existing approaches in network worm detection and principally behavior based approaches are not sufficiently accurate to perform network worm detection due to simple heuristics and inefficient approaches used to detect network symptoms. The intention of this thesis is to propose and implement an approach that detects the existence of network worms in the network.

The proposed approach is known as; A Statistical Approach towards Worm Detection using Cross-Relation Technique **(**ASAWDCRT) consists of three sub-algorithms, namely. Network scanning, network worm correlation and network worm signature automation. Network scanning approach, aims to detect TCP and UDP random and sequential scanning, and consists of three sub modules, including (i) filtering module (ii) traffic statistical analyzer module and (iii) cross relation module. Network worm correlation approach, aims to detect the destination source correlation behavior for network worm and consists of two sub modules, including (i) Destination port correlation based network worm detection module and (ii) Alert module .Network worm signature automation approach, which aims to generate a behavior signature for network worm symptoms (scanning and destination source correlation).

Network scanning algorithm is based on using TCP RST packets for detecting TCP sequential scanning and ICMP type 3 (port unreachable) packets for detecting UDP sequential scanning. TCP random scanning is confirmed when there is a 'cross-relation' between an ICMP type 3, code 1 (host unreachable) and the TCP RST. UDP random scanning is confirmed when there is a 'cross-relation' between an ICMP type 3, code 3 (port unreachable) and an ICMP type 3, code 1 (host unreachable).

Using ASAWDCRT, significant suppression gain of 94.75% in term of accuracy detection for TCP worm detection and 99.9% in term of accuracy detection for UDP worm detection as compared to the existing worm detection schemes. Meanwhile, scanning algorithm tested with the DARPA 1998 and NAv6 datasets

and confirmed that scanning algorithm was more effective in detecting TCP and UDP scanning than the existing algorithm, and it also provided better detection accuracy with average accuracy 66.6%.

CHAPTER ONE

INTRODUCTION

1.0 Review of Network Worms

Modern companies and organizations rely heavily on local computers, networks, and the internet to deliver and support their main business and operations. Thus, maintaining the health and security of the network is of great importance. In addition, sharing of network assets and devices makes these networks vulnerable to attacks.

Network threats are on the rise. Network worms are dangerous threats due to the speed of their propagation. Once a network worm infects a network, it will automatically begin to propagate, which will cause great destruction throughout the network due to network congestion. This will create unnecessary traffic, which serves only network worm propagation.

Ever-growing malware threats (such as network worms) are driving network security administrators to investigate solutions that can detect and protect their online Web and social media environments. These solutions include blocking inbound malware and analyzing outbound traffic to detect compromised endpoint systems.

The severity of network worms depends on the propagation process, wherein network scanning is initiated to determine the vulnerability of the host and services. Network scanning will degrade network performance and consume bandwidth and

resource (CPU and memory) by making the network machines busy due to the requests that are received and responded in the scanner machine.

1.1 Background

Network security aims to protect information from interceptions by intruders. However, intruders still successfully bypass network security by employing malicious codes and techniques. A common example of malicious code is the network worm. Network worm is a self-propagated standalone program; it does not require other programs for its propagation. The life cycle of a network worm after its release typically includes four phases: target finding or scanning, network worm transferring, network worm activation, and infection. The network worm is active on the network during target finding; network worm transferring and can be detected by network-based intrusion detection systems (NIDSs). The activities in the last two phases (activation and infection) are limited to local machines and are harder to detect by NIDSs because the network worm activities are more focused on individual computers rather than on the entire network. In contrast, the activities in the first two phases (scanning and transferring) are easier to detect because network worm activities are centered on the network, such as the existence of abnormal traffic generated from scanning. Figure 1.1 shows the typical location of NIDS in the network.

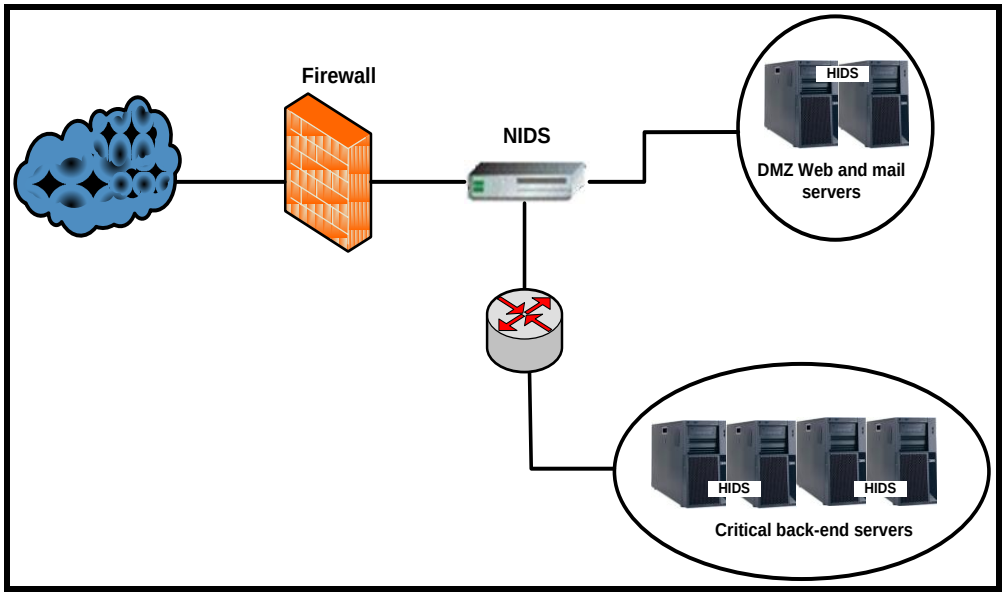

Figure 1.1: The typical location of NIDS in the network

1.1.1 Intrusion Detection System (IDS)

IDSs are typically used to detect intruder and anomalous activities within individual hosts. Network researchers such as Antonatos (2004) defined IDS as the process of determining whether the attack was an attack attempt or whether it actually took place.

An ID is deployed at the main gateway to monitor network traffic and to check the traffic packets against specific rules. These rules were developed to detect suspicious packets (such as network worm packets) or anomalous traffic (Chen, Cheng & Hsieh, 2009). IDSs are classified into two main categories: signature-based and anomaly-based detection systems (Zhou, 2006; Fisk, 2002; and Raghunath, 2008).

1.1.1.1 Signature-based Detection System

Similar to a computer virus or other malicious codes, a network worm has a signature that can be used by IDS in the detection phase. In this type of detection, IDS checks the incoming packet payload and system log files against the network worm signatures that are already stored in the IDS database. An alert will be triggered when a match is found. One advantage of signature-based detection it's ability to detect known attacks with high accuracy rate. However, signature-based detection systems cannot detect zero-day or newly released attacks. In addition, these systems must have previously defined network worm signatures for all possible attacks, which require frequent network worm signatures updates to keep the signature database up-to-date (Patcha & Park, 2007).

1.1.1.2 Anomaly-based Detection System

A profiling program will be created for the normal behavior of the network, which will be used by the anomaly-based detection system as a baseline. Deviation from this baseline will be treated as an anomaly or a possible intrusion (Chen et al., 2010).

The main drawback of the anomaly-based system is the duration of the training period spent on profiling the normal traffic behavior of the network. In addition, the training process must be repeated if changes occur in the training environment. This situation is known as the drift problem, which complicates the creation of a normal traffic profile. Figure 1.2 shows the block diagram of a signature (misuse) and anomaly detection system.

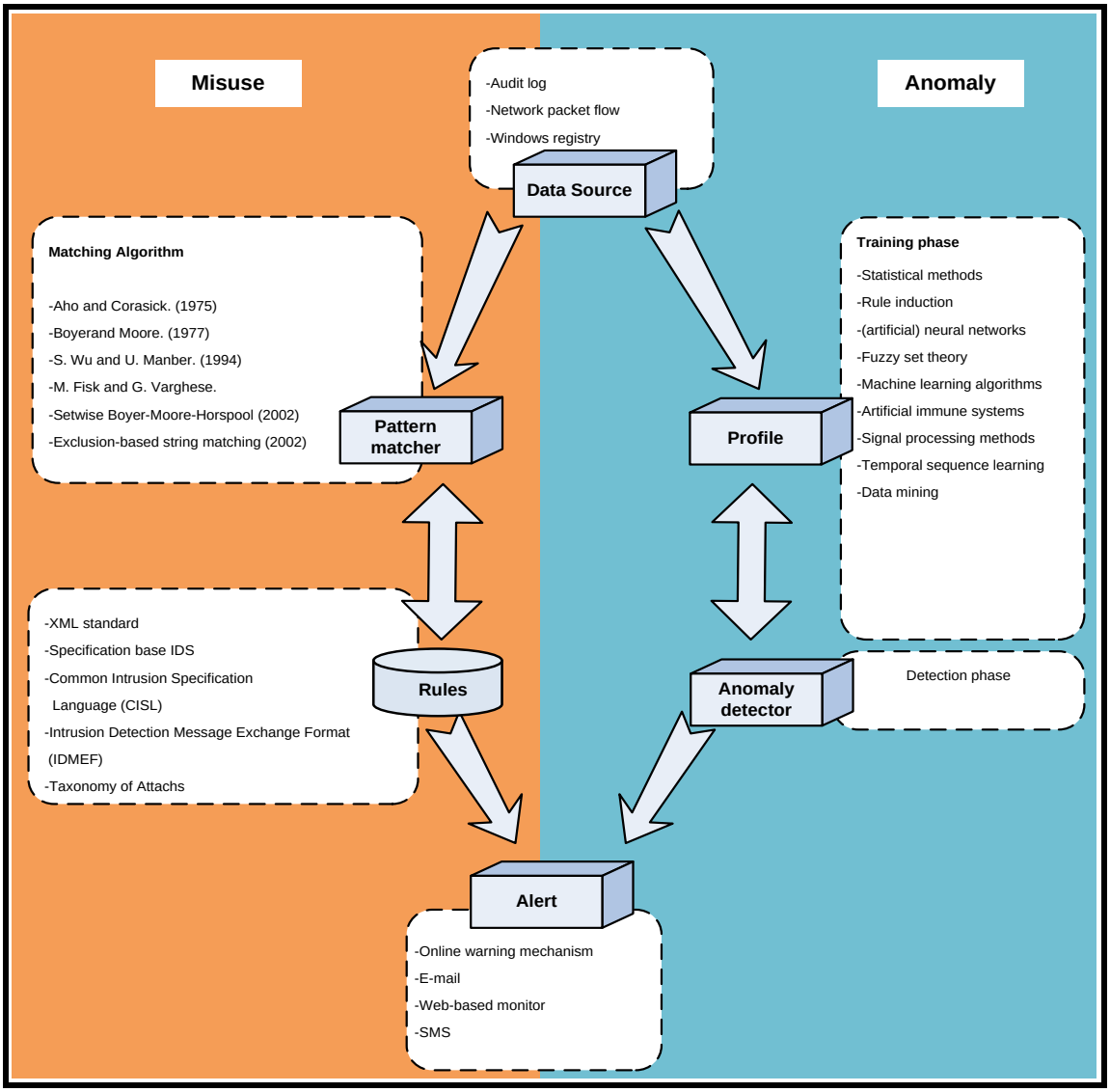

Figure 1.2: Block diagram of a misuse and anomaly detection system (Chen et al., 2010)

The proposed work is categorized as an anomaly-based detection technique. Specifically, it relies on the observation of network traffic to check the existence of the network worm based on network worm anomalies.

1.1.2 Network Scanning

Network worms conduct network scanning to determine the vulnerable hosts and services of a target network. Network scanning is considered to be the first step for attackers to gain access to the targeted network (Leckie & Kotagiri, 2002). This process allows unwanted network traffic into the network, which make the scanned hosts busy from receiving and responding to various unnecessary requests. Attackers are continuously adopting different scanning techniques for this purpose.

The most common scanning techniques are sequential scanning (such as TCP or UDP sequential scanning) and random scanning (such as TCP or UDP random scanning) (Li et al., 2008). In sequential scanning, the attacker scans a specific and known IP or a range of IP addresses sequentially, whereas random scanning enables attackers to select their target machines randomly.

Network scanning may exhibit traffic anomalies; because scanners may target the whole IP address range looking for hosts and vulnerable services. This is due to the fact that attackers are not aware of what services are available and vulnerable within the targeted network that they can use to initiate their attacks (De et al. 1999; Northcutt & Novak, 2002). TCP and UDP random and sequential scanning are commonly used by network worms to find out the vulnerable hosts and services

1.2 Problem Statement

Network worms are malicious codes that can propagate through the network without human intervention. Network worms are a challenging problem because of their highly destructive effects on network resources, topologies and services. The

entry point for network worms are the vulnerable hosts and services on the network. To locate vulnerable hosts and services, network worms launch a network scan, which is the first phase of the life cycle of a network worm. This process is followed by the transmission, activation and infection phase (Li et al., 2008).

Network scanning enables an attacker to gather information about his or her target, such as the operating systems, system architecture, and services that run on each computer. Network scanning is the first step for attackers to gain access to the target network. Identifying the information scanned by attackers can assist system and network administrators to determine the purpose of the attacks. Thus, resources and services can be further protected by patching or installing security measures, such as firewalls, IDS, and computer systems.

Several network scanning detection approaches have been proposed in the literatures such as anomaly based approach, statistical anomaly based approach and aggregation based approach, the best is statistical anomaly based approach (Zeidanloo et al., 2010). Examples of this approach are the Statistical Packet Anomaly Detection Engine (SPADE) proposed by Staniford et al., (2002) and the Statistical Scan Anomaly Detection Engine (SCADE) developed by (Freiling et al., 2005). However, it still has issues in term of accuracy as discussed in chapter two.

An attacker often blindly scans the network to find vulnerable hosts and services. However, they are not aware of the active IPs or the types of services that run on each host. Targeting inactive hosts or services will frequently generate connection failure messages, such as TCP RST, ICMP 3 code 1, and ICMP 3 code 3

packets (host and port unreachable). Frequent connection failures generated by network scanning might indicate the existence of a network worm if analyzed properly. As a result, several network worm detection techniques based on connection failure have been proposed in the literature, such as (Berk et al., 2003), (Jung et al., 2004), and (Yang et al., 2006). The basic idea of these approaches is to count a certain type of ICMP packet (such as ICMP type 3) within a specific period. An alert will be triggered if the number of ICMP packets exceeds the predetermined threshold.

In general, approaches that depend on connection failures have high false positive detection rates, which are attributed to two main reasons. First, sole dependence on connection failure to detect network worms is not sufficiently accurate because network worm behaviors exist in addition to connection failure. Second, several malicious codes that share the connection failure behaviors of network worms lead to misclassification and low detection accuracy.

Anomaly-based approaches have also been proposed for detecting network worms and generating the signature of unknown network worms. Among these approaches are the AutoGraph (Kim & Karp, 2004), EarlyBird (Singh et al., 2004), Anagram (Wang et al., 2006), and LESG (Li et al., 2007).

These approaches examine the content traffic of a network. However, these approaches cannot monitor encrypted traffic. Even when network worms are transmitted using unencrypted connections, advances in polymorphism research such as (Mason et al., 2009) have exposed the premise of these detectors. These

approaches generate signatures consisting of a single, contiguous sub-string of a network worm's payload, which match all network worm instances. Unfortunately, these signatures fail to match all polymorphic network worm instances.

As a result, a better solution for network worm detection is needed. Therefore, this thesis addresses the following issues:

1. The existing behavior-based approaches (such as DSC approach) for network worm detection with high false positive rates and low detection accuracy are examined. The factors that affect the accuracy of the entire system are addressed to minimize the rate of false positives.
2. The existing approaches for generating network worm signatures, which cannot sufficiently identify all polymorphic network worms, are investigated.

1.3 Research Objective

The objectives listed below are intended to solve the problem of low accuracy in terms of network worm detection. The main objective of this thesis is to propose a new behavioral based algorithm to increase the accuracy of network worm detection. The specific objectives of this thesis are as follows:

1. To propose a new statistical algorithm for increasing the accuracy of network scanning detection by using the cross-relation technique.
2. To generate a behavioral signature for network worm symptoms [scanning and destination source correlation (DSC)].
3. To test and evaluate the proposed behavior based network worm detection and compare it with existing behavior based approaches

1.4 Research Contribution

The current behavioral based approaches, which are used to detect the presence of network worm in the network, are not accurate in detecting network worm. We propose a new algorithm called A Statistical Approach towards Worm Detection using Cross-Relation Technique (ASAWDCRT), which is designed to detect network worms with improved accuracy and efficiency. If detects the presence of network worm in the network in the early stage provide the network administrator with the advantage of taking action before the others network machines are compromised. High accuracy can be achieved by using advanced approaches to detect network worm anomalies that appear in the first two phases of network worm life cycle (scanning and network worm transferring). The detailed contribution of the present work as the follow:

The proposed ASAWDCRT provides the following key features:

a) A new statistical algorithm that increases the accuracy of network scanning detection, which is important because network scanning is the first step of network worm propagation.

b) A new behavior-based algorithm for network worm detection. This approach uses the proposed scanning approach to detect infected hosts that perform scanning.

1.5 Research Scope and Limitation

The research scope is limited to the detection of network worms that propagate by employing TCP and UDP sequential and random scanning. The efficiency of the proposed scanning approach is decreased when the ICMP error messages are blocked or dropped by border routers or gateway systems.

CPU and memory complexity have a linear correlation to the number of packets captured and stored in the log table. To circumvent CPU and memory saturation, high-end server architecture can be employed to conduct packet capture and analysis, while maintaining low time consumption.

The dataset, which is used for testing the behavior-based network worm approach, was generated by GTNETS simulator. Behavioral signature automation was conducted for TCP and UDP sequential and random scanning and DSC behaviors.

1.6 Research Methodology

The proposed algorithm presents A Statistical Approach towards Worm Detection using Cross-Relation Technique. This approach is known as ASAWDCRT (Figure 1.3).

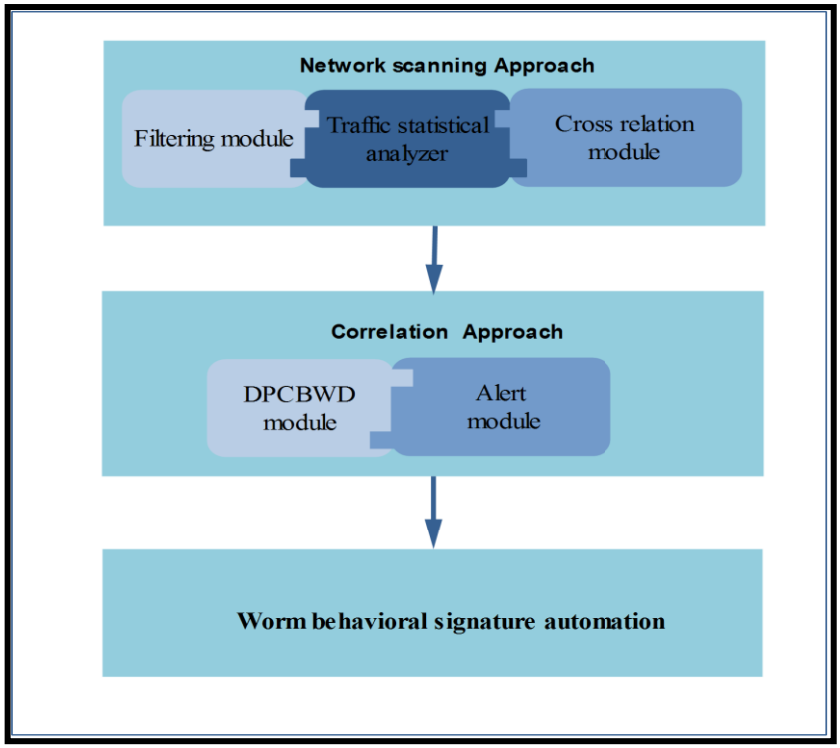

Figure 1.3: The Proposed Algorithm

Figure 1.3 shows that ASAWDCRT comprises the following three algorithms:

1. Network scanning s aims to detect TCP and UDP random and sequential scanning, which consists of three sub–modules: (1) filtering module, (2) traffic statistical analyzer module, and (3) cross-relation module.
2. Network worm correlation algorithm detects the Destination Source Correlation (DSC) behavior for network worm, which consists of two-sub modules: (1) destination port correlation-based network worm detection module (DPCBWD), and (2) alert module.
3. Network worm signature automation algorithm generates a behavior signature for network worm symptoms (scanning and DSC behaviors).

1.7 Thesis Organization

This thesis is organized into six chapters. Chapter 1 presents the objectives of this thesis, which starts by presenting a background discussion for network worms, network scanning and IDS. The research objectives, contributions, scope and limitation, and research methodology are also provided in this chapter.

Chapter 2 discusses the most current and related works in behavior-based network worm detection, scanning approaches, and signature automation approach. The advantages and drawbacks for each approach are highlighted.

Chapter 3 presents the methodology and the design of the proposed solution. Statistical "Cross-Relation" Approach for detecting TCP and UDP random and sequential Network Scanning (SCANS) is introduced in this chapter, as well as the behavioral signature automation algorithm.

Chapter 4 covers the implementation details of ASAWDCRT, database structure of ASAWDCRT and capture engine.

Chapter 5 provides an in-depth analysis and dissuasion for ASAWDCRT approach, this chapter is divided into two parts. The first part reports the results of accuracy detection for ASAWDCRT. The second part reports the result of comparative test between ASAWDCRT and DSC algorithms.

Chapter 6 provides the conclusion and recommendations for future work.

CHAPTER TWO

LITERATURE REVIEW

This chapter presents background information and applicable research related to network worms, network scanning and network worm automation. Section 2.1 provides a brief introduction to network worm and network worm life cycle. Section 2.2 relates the network worm transmission schema, while Section 2.3 provides a deeper discussion on network worm detection approach. Section 2.4 presents the approaches that are used in the network scanning detection. Section 2.5 presents the approaches that are used in the network worm's signatures automation.

2.1 Network worms

A network worm is a self-propagating, self-duplicating malicious code that spread without human intervention in computer networks and attacks vulnerable hosts and services. Network worms are typically classified based on two attributes: methods used to spread and the techniques used to exploit vulnerabilities. Meanwhile, network worms have destructive effects in the network topology, resources and service. Therefore, many researchers attention have been employed to propose techniques to detect the presence of network worms in the network (Cohen et al., 2008). Table 2.1 shows propagation scheme and exploits for different network worms.

Table 2.1: The propagation scheme for different network worms

Year	Name	Exploits	Propagation Scheme
1998	Morris	Vulnerabilities in UNIX Send mail, Finger, sh/rexec; weak passwords	The network worm harvests hostnames from local files and sends object files to a target machine. The target then opens a connection back to the originator, which creates a duplicate process in the target machine.
2001	Code Red	MS IIS vulnerability	The network worm sends a malicious HTTP payload to a randomly generated IP address on the TCP port 80.
2001	Nimda	MS IE and IIS vulnerabilities	Nimda sends itself by email or copies infected files to the open network shares and to vulnerable MS IIS Web servers via TFTP on the UDP port 69.
2003	Slammer	MS IE and IIS vulnerabilities	The network worm sends a malicious UDP packet to a randomly generated IP address on the port 1434.
2003	Blaster	MS IE and IIS vulnerabilities	The network worm attempts to connect to a randomly generated IP address on the TCP port 135. Successful attack starts a shell on port 4444 through which the originator Instructs the target. The target downloads the network worm using the originator's TFTP server on the port 69
2004	Witty	Internet Security Systems software vulnerability	The network worm sends a malicious UDP packet From the source port 40000 to randomly generated IP addresses.
2005	Zotob	Internet Security Systems software vulnerability	The network worm attempts to connect to a randomly generated IP address on the TCP port 445. Successful attack starts a shell on port 8888 through which the originator instructs the target. The target downloads the network worm using the originator's FTP server on the port 33333.

2.1.1 Network Worm Life Cycle

According to Li, Salour & Su (2008) network worms life cycle consist of target finding, network worm transferring, network worm activation and infection.

2.1.1.1 Target Finding (Network Scanning)

Target finding or network scanning is considered the first step in the network worm life cycle, a network worm tries to explore the computer networks to find out the vulnerable hosts and services to infect. In this phase, the characteristic behaviors for the network worms are obviously appearing and can easily be detected by intrusion detection systems (IDS) (Li, Salour, & Su, 2008). Network worm uses different type of network scanning techniques to find out the victim. The scanning techniques are categorized into five categories as shown in Figure 2.1**.**

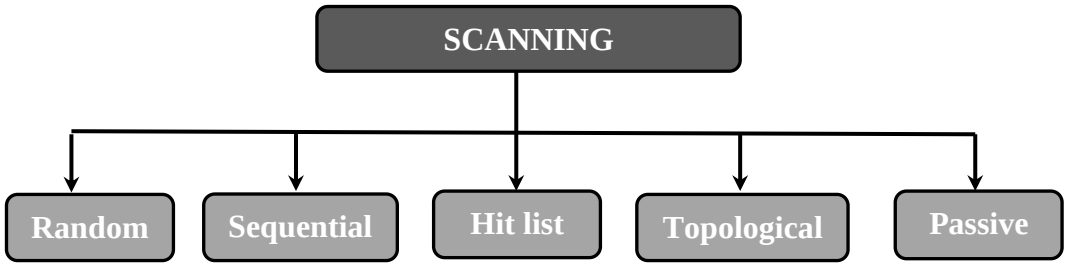

Figure 2.1: Scanning techniques methods (Li et al., 2008).

A. Random Scanning

In this method, an attacker often blindly scans the network to find out the vulnerable host's and services. However, he is not aware which IP is active or what services are running on each running host. On the other hand, targeting inactive hosts or services will frequently generate connection failures messages. Thus frequent connection failures generated by network scanning might indicate the existence of a network worm if it is being analyzed properly (de Vivo et al., 1999; Northcutt &

Novak, 2002). The connection failure may occur due to the following: (1) a network worm tries to scan some services but the port is closed; in this case the ICMP port is unreachable or TCP RST packet will be generated (2) a network worm tries to scan an inactive host; in this case the ICMP host is also unreachable and a packet will be generated. Random scanning is common scanning technique employed by network worms. This thesis focus on detecting TCP and UDP random scanning

B. Sequential Scanning

In this method, the attacker aims to scan a block /range of IP addresses sequentially. After the network worm has randomly selected a starting IP (s), a scanner will continue to scan s+1 (such as 10.207.161.3, s+1=10.207.161.4) or s-1(such as 10.207.161.3, s-1=10.207.161.2) (Gu, et al., 2004). Sequential scanning can be easily noticed by any traffic sniffing tool such as Wireshark. This is because the captured IP that is performing the sequential scanning is listed sequentially. This thesis focus on detecting TCP and UDP sequential scanning.

C. Hit List Scanning

In this method, an attacker defines a list of vulnerable hosts and services to be scanned once the network worm is released. This list may be generated by stealthily monitoring the network or from somewhere else. The accuracy of this method is high since the attacker has prior knowledge about the targets and services. Due to its high accuracy, the probabilities of anomaly behavior that may appear is very low, so it is difficult for anomaly detection systems to detect such kind of scanning (Chan et al., 2006).

D. Topological Scanning

In this method, a network worm based on the local information is saved into the hosts. Local information includes email address in user contact list, host file (e.g., /etc/hosts) and URLs in the user's browsing history. Attackers will employ this information to identify its targets and infection path by using a second channel such as the services provided by Google, or by querying a peer-to-peer network or an instant messaging server for vulnerable peers. Topological network worms can spread very fast, especially on networks with highly connected applications (Weaver et al., 2003).

E. Passive Scanning

In this method, the information about vulnerable hosts and services are obtained by monitoring the target network passively (Kato et al., 1999). This form of scanning is much slower than the previous techniques but can be harder to detect by intrusion detection systems (IDS) since it does not appear to show any anomalies behavior

2.1.1.2 Network Worm Propagation Schemes

As reported by Weaver et al. (2003), there are three network worm propagation schemes which are as follows (1) self-carried (2) embedded (3) second channel. Self-carried network worms are actively transmit itself to the target host (the network worms are fully transmitted to the target during the initial connection), second channel network worms need second communication channel, in this scheme, the network worm communicates with the victim machine using original channel then the victim machine connects back to the infecting machine using another channel to download the network worm payload. The embedded propagation scheme is very stealthy and it's done by append the payload after, or replace, legitimate traffic to

hide itself. No anomalous events will be triggered, and it is hard for anomaly-based detection systems to detect. In addition to the three propagation schemes discussed, botnets have been utilized to propagate network worms, spams, spyware, and launching distributed denial-of-service (DDoS) attacks (Li et al., 2008). A botnet is a group of compromised hosts under the control of a botmaster. The communication channel for the botmaster to issue commands that can be implemented using different protocols such as http or point-to-point (P2P) protocols.However, the majority of botnets use the Internet Relay chat (IRC) protocol for this purpose (Gu et al.,2008; Fabian et al., 2007).

2.1.1.3 Network Worm Activation Schemes

Network worm activation means running network worms under certain condition or schedule. Weaver et al. (2003) classified the network worm activation network worm as following:

1. Human activated, this type of network worm activation require human intervention to execute the network worm
2. The network worm activate based on specific activity which is performed by user (such as open CD or bin drive)
3. Activated by a scheduled processes. In scheduled process, network worms are activated by a legitimate automated process which has not been properly secured, such as a legitimate program which automatically updates itself from an infected web server.
4. Self-activated, this kind of network worm can activate without human intervention, and it considers the most dangerous one, this thesis focus on detecting this kind of network worms.

2.2 Approaches Used in Network worm Detection

There are many approaches proposed for network worm detection. In the following, the commonly used anomaly approaches which are used to detect network worms based on an artificial neural network and connection failure approaches.

2.2.1 Artificial Neural Network (ANN) Based Network Worm Detection

ANN is an information processing paradigm that is inspired by the way biological nervous systems (i.e., the brain) are modeled with regard to information processing. The key element of this paradigm is the structure of the information processing system. It is a network composed of a large number of highly interconnected processing elements, called neurons, working together in order to approximate a specific function (Moskovitch, Elovici, & Rokach, 2008)

Stopel et al. (2006a) proposed an approach for detecting infected host by network worms based on ANN. This approach used the infected host resources such as CPU and memory in the network. In addition, the study utilized feature selection techniques for the dimension reduction, the used selection techniques are as follow (1) The relation between the inputs and the hidden neuron's relative variance (2) The Fisher score ranking (3) Gain Ratio Filter. The average accuracy for the proposed approach was 99.98 %.

The outputs of these techniques are the features that have impact in computer behavior which are infected by network worms. The study evaluated each technique by preprocessing the dataset accordingly and training the ANN model with the preprocessed data. Furthermore, the ability of the model to detect the presence of a

new computer network worm was evaluated. In particular, during heavy user activity on the infected computer.

Another research proposed by Stopel et al. (2006b) adopted ANN and two other known classifications techniques, Decision Tree and k-Nearest Neighbors, to observe their ability to classify computer network worms during heavy user activity on the infected computers. In this study, a number of computers infected with a different number of network worms and different parameters distributed in the various measurements such as processor features, TCP layer features, UDP layer features, IP layer features and low Network Interface features. Moreover, the study evaluated each technique by preprocessing the dataset by training the ANN model with the preprocessed data. The average accuracy for the proposed approach was 85.0 %.

The proposed approach in (Stopel et al., 2006a; Stopel et al., 2006b) detects malicious activity of network worms by looking at the attributes derived from the computer operation parameters such as memory usage, CPU usage, and traffic activity. The main drawback of this model was appearing in misclassifications of network worms in the beginning of their activity. Meanwhile, observing all computer features in the network are time and recourse consuming.

Farag et al. (2010) proposed a method for detecting unknown network worms based on local victim information. The proposed method initialized an ANN for classifying network worm / non-network worm traffic in every host. The traffic classification was performed by using two models which are Classification

Prediction Combined model (CPC) and Classification Prediction Separated (CPS) model. In CPC the goal was to use ANN to produce two outputs (network worm traffic and percentage of infection). In CPS model, two ANN networks were used to solve the classification problem. To evaluate the proposed approach, a simulated dataset was adopted and the output generated a reliable result with accuracy of 99.96% in detecting the presence of network worm over the network, even for unknown network worms.

The ANN approach has computational advantages when real-time computation is needed, and has the potential to detect previously unknown network worms with high level accuracy. Also, ANN has advantage to reduce the feature dimensionality. However, the two shortcomings for ANN techniques are

(1) Training period (takes time) and

(2) Involvement problem (any changes in target environment will affect the training dataset).

2.2.2 Connection Failure Based Network Worm Detection

The connection failure in the network appears in the form of ICMP Type 3 (port unreachable), ICMP Type-3 (destination unreachable) and TCP RST packets. The existing of these packets in a high rate means that there are many connection failures which are considered as very strong footprint and symptom for network scanning (first stage in network worm life cycle). An ICMP Type3 code1 (host unreachable) packet is generated when TCP/SYN or UDP packet is sent out to an unused IP address. Meanwhile, ICMP Type3 code 3 (port unreachable) packets are

generated when TCP/SYN or UDP packet is sent out to an existing address but the port closed (Li et al., 2008; Rasheed et al, 2009).

The TCP RST packet is generated in two cases, when a TCP-SYN packet is sent out to an existing host but the port is closed, and when a TCP-SYN carries forged source IP address that is send to an existing host, the destination host will reply with SYN/ACK packet to the real host, in this case, the TCP RST packet is send from a real IP address to the destination host as depicted in Figure 2.2 (S. Chen & Ranka, 2005).

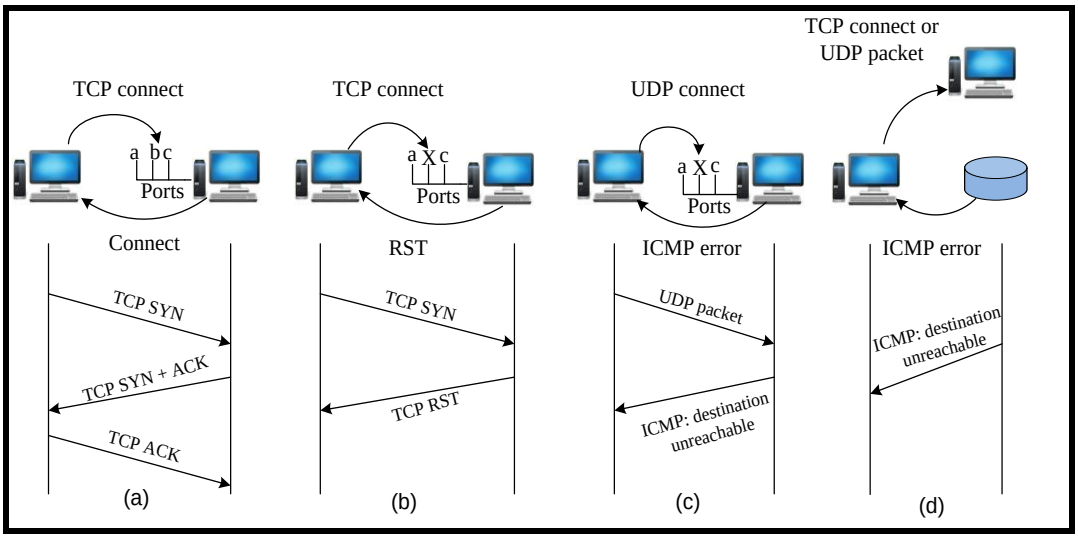

Figure 2.2: Connection attempts: a) successful TCP connection b) TCP destination port closed c) UDP destination port closed, d) destination IP address does not existing.

Chen & Tabg (2007) analyzed the essential characteristics of TCP based network worm propagation that concerned on sending a large number of TCP connection requests and developed a Distributed Anti-network worm System (DAW) to detect and contain the network worms based on the received failure requests through the routers. New defense techniques are developed based on the difference between normal hosts and network worm-infected hosts. In fact, a network worm-

infected host has a notable high connection-failure rate when it randomly scans the internet.

This property allows DAW to set the network worms apart from the normal hosts. The proposed defense techniques are the temporal rate-limit algorithm and the spatial rate-limit algorithm. Temporal rate-limit algorithm is designed to constraint the maximum number of failed requests daily (constraints done by determining the normal number of connection failure). On the other hand, spatial rate-limit algorithm constrains the combined scanning rate of all infected hosts in the network. The purpose of this approach is to slow down or even halts the network worm propagation within Internet service provider (ISP). The drawbacks of DAW are as follow (1) unable to detect network worm using UDP transmission schema (Li et al., 2008) such as slammer (Moore et al., 2003) (2) unable to detect low rate scanning network worm (Rasheed, Norwawi, Ghazali, & Kadhum, 2009). The effectiveness of the new techniques is evaluated analytically by simulation. Figure 2.3 shows the DAW architecture.

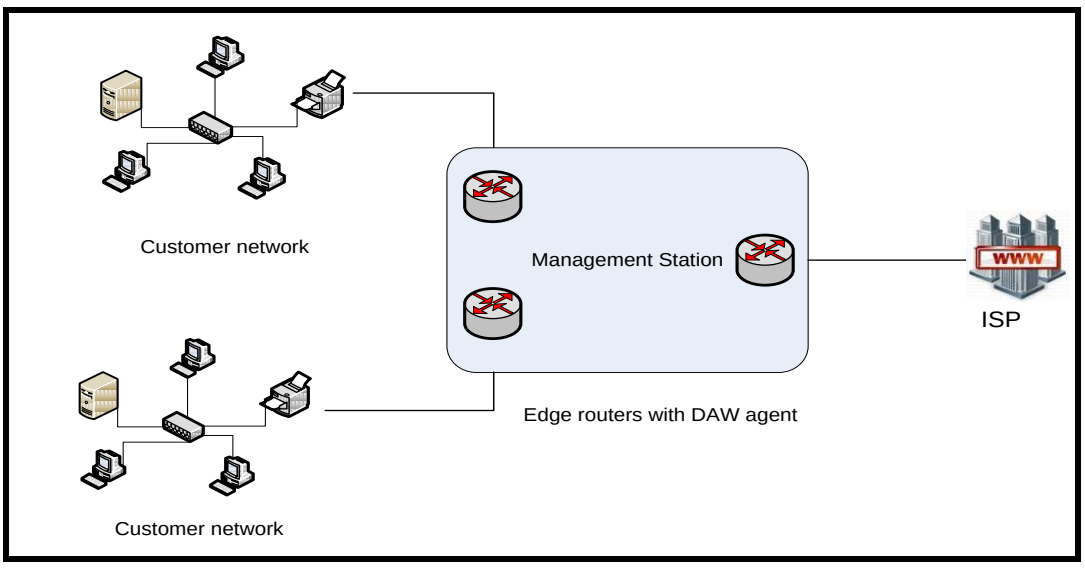

Figure 2.3: DAW architecture (S Chen & Tang, 2007).