DDoS ATTACK DETECTION USING FREQUENCY DOMAIN CHARACTERISTICS

by

Ramin Fadaei Fouladi

B.S., Electrical and Electronics Engineering, Amirkabir University of Technology
(Tehran Polytechnic)-Iran, 2004

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in FBE Program for which the Thesis is Submitted
Boğaziçi University
2014

DDoS ATTACK DETECTION USING FREQUENCY DOMAIN CHARACTERISTICS

APPROVED BY:

Prof. Emin Anarim ....
(Thesis Supervisor)

Assoc. Prof. Mehmet Akar ....

Prof. Fatih Alagöz ....

DATE OF APPROVAL: 09.01.2014

ACKNOWLEDGEMENTS

I would like to express my deep gratitude to Prof. Emin Anarim, my research supervisor and my advisor, for his encouragement, invaluable guidance, and support of this research work. I would like to express my great appreciation to Assoc. Prof. Güneş Karabulut Kurt, for her constructive advice and for her willingness to give her time so generously. I am also thankful to Prof. Fatih Alagöz and Assoc. Prof. Mehmet Akar for their suggestions to improve this work.

My special thanks should be given to my friends and colleagues, Tina Seifpoor, Cemil Eren Kayataş and Ayda Sadeghzadeh, for their support all through this thesis. I am also thankful to Derya Erhan and Çağdaş Şenel for their contribution to this project. I would like to extend my thanks to the staff of the Electrical and Electronics Department for their help in offering me the resources in running the program.

Finally, and very importantly, I wish to thank my beloved mother, Akhtar Narenjiyeh, and my brother, Daniel F Fouladi, for their support in all aspects of my personal and academic life.

ABSTRACT

DDoS ATTACK DETECTION USING FREQUENCY DOMAIN CHARACTERISTICS

Providing 24-hour service to the users, is one of the major concerns of network administrators. A denial of service attack refers to a condition that a server cannot give normal services to its legitimate clients due to the large amount of bogus packets sent by an unknown source. In a distributed denial of service (DDoS) attack, an attacker launches the attack on a server via a large number of unaware computers through Internet. During a DDoS attack, the victim is forced to reply to the requests from those infected nodes called zombies. The first step of countermeasure against these types of threats is detection. Conventional methods analyze the contents of packets arrived to the victim node to find an abnormality. Although they can identify some simple attacks, they are almost unable to segregate the source of normal traffic from attack one when attackers alter the source IP address into the normal source IP address. Additionally the contents of the abnormal packets are usually changed intentionally by attackers to be close to those in normal packets and therefore they can easily be passed through a system employing traditional detection approaches. In this thesis, a frequency domain analysis is proposed to detect DDoS attacks. The number of packets received by the victim in a specific interval are sampled and considered as a random process. Employing two different methods of power spectral density estimation, the frequency characteristic of the time series is estimated. Using each spectrum estimation methods, two sets of frequency characteristics, one for normal and another for DDoS traffic, are acquired, and utilized by a signature based intrusion detection system to detect abnormality.

ÖZET

FREKANS TABANINDA DDoS SALDIRI TESPİTİ

Kullanıcılara 24 saat servis vermek ağ yöneticilerinin esas problemlerinden biridir. Servis engeleme saldırıları sunucuya gelen büyük oranda sahte paketler yüzünden sunucunun yasal kullanıcılara servis verememesi durumuna dayanmaktadır. Dağıtık servis engelleme saldırılarında, saldırgan internet üzerindeki çok sayıda habersiz bilgisayar aracılığı ile bir sunucuya doğru saldırı başlatır. DDoS saldırısı sırasında hedef bilgisayar, virus bulaşmış uçlardan gelen isteklere cevap vermeye zorlanır. Bu tür tehditler karısında ilk adım saldırının tespitidir. Geleneksel yöntemler, hedef bilgisayara gelen paketlerin içeriği bir anormallik bulmak için incelenir. Bu yöntemler, basit saldırıları tespit edebilmelerine rağmen saldırganlar, saldırı için normal kaynak IP'lerini kullandıklarında, saldırının kaynağını normal kaynaklardan ayırt etmekte yetersizdirler. Ek olarak olağandışı paketlerin içeriği genellikle saldırganlar tarafından normal paketlerle benzer olmaları için bilinçli bir şekilde değiştirilirler. Bu paketler geleneksel saptama yaklaşımlarını kullanan bir sistemden kolaylıkla geçebilirler. Bu tezde, frekans bölgesi incelemesi ile DDoS saldırılarının tespiti önerilmiştir. Belirli bir zaman aralığında hedef bilgisayar tarafından alınan paketlerin sayısı örneklenerek rastgele süreç olarak düşünülmüştür. İki farklı güç spektrum yoğunluk yöntemi kullanılarak, zaman serisinin frekans karakteristiği kestirilmiştir. İki spektrum kestirim modeli kullanılarak, bir normal bir DDoS trafiği olmak üzere iki farklı frekans karakteristiği elde edilmiş ve bu imza tabanlı bir saldırı tespit sistemi tarafından anormallik tespiti için kullanılmıştır.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF SYMBOLS

$X[n]$	Time series
$u[n]$	White noise
p	Order of AR model
q	Order of MA model
r	Autocorrelation coefficient
R	Autocorrelation matrix
e_n^+	Forward prediction error
e_n^-	Backward prediction error
V_n	Eignevector
$\hat{P}$	Pseudo spectrum
D	Overcomplete dictionary
E	Estimation error
$R_x^k[n]$	Residual after k iterations
d_k	A column of overcomplete dictionary
$d_{\gamma i}$	*ith* matched atom
a_γ	Corresponding cross-correlation of $d_{\gamma i}$
μ	Expected value
σ^2	Variance
ρ_k	Autocorrelation function
ε_n	Total prediction error
λ_n	Eigenvalue

LIST OF ACRONYMS/ABBREVIATIONS

ACF	Autocorrelation Function
AR	AutoRegressive
ARMA	AutoRegressive Moving Average
CSV	Comma Separated Values
DDoS	Distributed Denial of Service
DFT	Discrete Fourier Transform
DoS	Denial of Service
FFT	Fast Fourier Transform
FPE	Final Prediction Error
FTP	File Transfer Function
HIDS	Host-based Intrusion Detection System
HTTP	HyperText Transfer Protocol
ICMP	Internet Control Message Protocol
IDS	Intrusion Detection System
IP	Internet Protocol
IPS	Intrusion Prevention System
IRC	Internet Relay Chat
K-SVD	K-mean Singular Value Decomposition
LTI	Linear Time Invariant
MA	Moving Average
MUSIC	Multiple Signal Classification
NCS	Normalized Cumulative Spectrum
NGMN	Next Generation Mobile Network
NIDS	Network-based Intrusion Detection System
PSD	Power Spectral Density
RTT	Round-Trip Time
SVD	Singular Value Decomposition
TCP	Transmission Control Protocol
UDP	User Datagram Protocol

VOIP	Voice Over IP

1. INTRODUCTION

Nowadays, Internet has become an indispensable part of human life. A vast variety of applications such as paying bills, mobile networks, VOIP, On-line games, video conference and so on, are used through Internet by people. Being without Internet even for a short time is frustrating; therefore, securing the availability of web servers, those provide services to their legitimate users, is the main concern of network administrators. Among numerous threats present on Internet, the one that menaces the availability of a website is denial of service attack (DoS). A DoS attack makes a website out of reach by flooding its resource or bandwidth. The most harmful types of DoS attacks are distributed denial of services (DDoS). In a DDoS attack, large number of attackers invade to the server, and make it down.

Diagnosing DDoS attack is the main problem of detection systems. Conventional detection methods use packet-level analysis to detect abnormality in the network. Although these approaches can find abnormality to some degree, but if the attackers spoof the source IP addresses, they are ineffective. In addition they cannot differ between DoS and DDoS attacks. Therefore, a new method which identifies and classifies different attacks is necessary. Working in the frequency domain instead of the time domain would be an alternative. Considering the number of packets arriving to the server as a random process, transferring it to the frequency domain and extracting the frequency characteristics may help to distinguish between normal and abnormal traffics.

In this thesis, we will exploit the frequency characteristics of normal and DDoS traffics. Then we will employ them in a detection system to find abnormality in the network. The rest of this thesis is organized as follow: in the second Chapter we will discuss the DDoS attacks in detail. Chapter 3 is devoted to frequency analysis. Two different methods of spectrum estimation as well as previous works regarding to frequency analysis will be considered in this chapter . In Chapter 4, we will show a sparse representation method. The results will be discussed in Chapter 5 and finally Chapter 7 will conclude this work.

2. DENIAL OF SERVICE ATTACKS

DoS and DDoS are considered as major problems in Internet and networks. In these kinds of attacks, an attacker tries to make a server unavailable for its legitimate users. This job is accomplished by either depleting the resource of the server or by sending a large amount of illegal requests to the server. Users and their webservers usually interact with each other through sending some legitimate packets and requests such as http request. Each web server has a threshold value for the number of requests per second. If the number of demands goes over the threshold, the server will be incapable of handling all of them, and as a result the webserver goes into a non-active mode where it will deny any requests. This situation is called denial of service, and at this moment the server is out of reach from legitimate users. When a webserver is about to be born, the number of users, and the host capacity are estimated. Therefore each webserver has its own innate limitation.

To accomplish a DoS attack, the attacker makes use of a software to send multiple http requests to desired webserver trying to prevent its legitimate service. As a result, authentic users cannot reach to their server. If you want to connect to a web which is under the attack by a DoS attack, you will encounter with timeout error in your browser. These two attacks get their name based on the source of attack. If there is just one single computer contributing in the attack, the attack called DoS attack; on the other hand, if there are several systems and computers participating in the attack, the attack is called DDoS attack [5].

Those computers and systems which are used in DDoS attack are usually unaware of contributing in the attack. These innocent computers are referred to zombies or agents. The attacker exploits them by finding vulnerabilities on their systems [6]. To synchronize all systems, the DDoS attacker installs attack software on these computers through a secure channel. This software is responsible to coordinate all systems together to run an effective offense. During the attack all compromised systems begin to send useless packets toward victim simultaneously. Suddenly, the victim faces with a large volume of malicious traffic which it is unable to handle all at the same time [7].

There are some characteristics which can be used to differentiate denial of service attacks from other types of insulting and hacking. During an attack, the server undergoes of a large volume of unusual traffic. The source IP address is usually spoofed (fake IP address), and the source or destination port is assigned randomly depending the type of attack [7]. The attacker must select the type of protocol of attack before launching it. The protocol can be TCP, UDP or ICMP; therefore, when there is a denial of service attack running on a victim, these protocols are dominant among others in the traffic. In contrast to most of the hacking processes which can be traced back very easily, the denial of services are very hard and time consuming to trace back to find the source of the attack because of the source IP spoofing and employing of several compromised system to make the attack possible. To decrease the possibility of tracing back and identification the origin of the attack, the attacker usually employs a great number of systems. He scans and intrudes the targeted system; finds a vulnerability in the system and installs the attack software on it.

Network administrators usually employ intrusion detection systems (IDSs) to cope with or to mitigate the effect of denial of service attacks. IDSs are those systems used by network authorities to find the abnormalities in the traffic. Although different methods have been invented to settle the problem of denial of services, but on the other side, attackers have also developed their tools to run very complicated type of attacks.

2.1. DDoS Attack Architecture

According to the methods of implementation, DDoS attacks are categorized into three different approaches: the agent-handler model, the Internet relay chat (IRC)-based model and reflector model [1].

2.1.1. Agent-Handler Model [1]

Figure 2.1 displays the block diagram of Agent-Handler model. It comprises of three main parts: client, handler and agents [1]. The attacker exchanges information with the rest of the attack system via client. The duty of the client is to make connection possible between the attacker and other members. The client communicates with agents and conveys the directions of the attacker to them by means of software packages located through the Internet called handlers. Sometimes master and daemons are used instead of the terms handler and agents respectively.

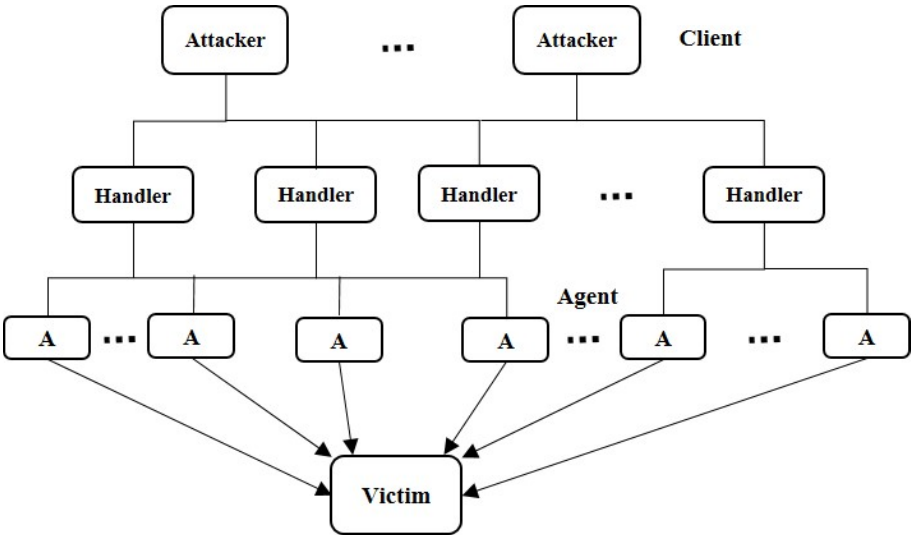


Figure 2.1. Agent-Handler model (reprinted from [2]).

2.1.2. IRC-based Model [1]

This model is similar to the previous one but, the handlers are substituted by Internet relay chat (IRC) systems. IRC is a free on-line chatting system that provides conversation environment for users siting before their computers [8]. The attacker takes advantages of the existence of these systems on Internet to communicate with its agents. Figure 2.2 displays this concept. The advantage of this method in comparison with the agent-handler method is that the IRC provides attackers with additional benefits. The system supplies the attacker with legitimate IRC ports for sending commands to the agents. Using these ports makes tracing the DDoS command packets more difficult.

Moreover, due to the large volume of traffic in IRC servers, the presence of the attacker in the system is imperceptible. Another advantage of IRC based system is that the attacker can log on to the IRC system and see a list of all available agents [9], instead of keeping the track of available agents, .

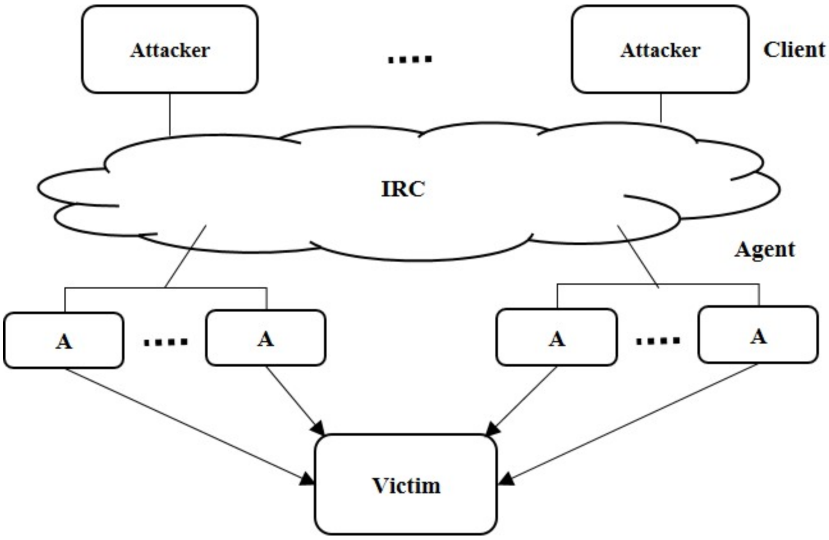


Figure 2.2. IRC-based model (reprinted from [2]).

2.1.3. Reflector Model [1]

Attacker, handler, agent, and reflector are the four main parts of this model. The principle of the attack is slightly similar to the previous ones. The difference is in the way of attacking to the victim. Instead of assailing directly to the victim, the attacker sends commands to the agents to forward packets with the victim IP address as the source IP address to other uninfected computers over the Internet. These packets force those healthy systems to answer to the traffic by sending return response. Because the source IP addresses in the packets are spoofed to the victim's IP address, all responses will be directed to the victim. A reflector can be any host on the Internet which responds to the requests. The attacker can also amplify the effect of this model by sending the packets to the broadcast address of the reflector networks [10]. A message sent to the broadcast address is received by all hosts attached to that network. In this way the attacker multiplies the volume of packets directed to the victim and has a severe impact on the performance of the victim's server.

2.2. Degree of Automation

The amount of attacker's intervention in the process of initializing the attack categorizes DDoS attacks into three different groups: manual, semi-automatic, and automatic. In the manual model, the attacker scans all possible vulnerabilities of the agents' systems, penetrates into the system and finally embarks on the attack. In semi-automatic, the attacker just commands the start of the attack. Handlers are employed to manage agents. Handlers coordinate agents together and mange the time and type of the attack. Having initialized, the attack will start. In automatic methods all processes from choosing, and synchronizing of the agents to starting the attack are done automatically. Because there is no sign of attackers in the automatic model, this type of DDoS model is very hard to trace back.

2.3. DDoS Attack's Target

According to the target of the attack in the victim system, the DDoS attacks are further put into two groups: bandwidth depletion and resource depletion attacks.

2.3.1. Bandwidth Depletion

The bandwidth of the victim is consumed by the unwanted packets filled by garbage data during a bandwidth attack. There is no further free bandwidth to handle the demands from legitimate users, and as a result, the victim becomes unavailable for its clients. Two major approaches are usually implemented in this method. In the first one which is called as flood attack, the victim is flooded by a large amount of UDP or ICMP packets. In the second one, known as amplification attack, packets, filled by victim's IP address as the source IP, are sent to the broadcast IP address. All clients in that network will receive these packets and send back the reply to the victim. So, the attack is amplified by the number of clients in the network.

2.3.2. Resource Depletion

In this approach, the attack is performed by sending malformed packets. Upon receiving a crooked packet, the server allocates some available resources to handle it. Because the packet is not healthy, it misleads the system and forces the system to allocate more resources to resolve it. The victim is engaged in coping with the problem occurring in the packets and therefore can not respond appropriately to the demands of normal users.

2.4. Intrusion Detection Systems (IDS)

To avoid suspicious activities and to monitor the traffic running in a network, IDSs are employed by the network administrator. If an abnormal event is spotted by the IDS, it will alert the administrator or it can takes action by itself to confront with that activity. The latter is applied in a specific type of IDSs called intrusion prevention system (IPS). An IDS is placed either in a host in the network or in a strategic section of the network, the former is called host-based IDS (HIDS) and the later is known as network-based IDS (NIDS). According to the method of detection, IDSs are grouped into two categories: signature-based and anomaly-based. In the signature-based, all incoming packets are compared with a database of attacks' signatures created by using the previous incidents. If the packets matches with a signature in the database, it will be considered as an attack. In anomaly-based IDS, the system models the legitimate traffics and if an incoming traffic violates this pattern, it will be considered as an attack.

3. FREQUENCY DOMAIN ANALYSIS

To analyze the variation of a signal relating to the different frequency components instead of using time domain, we map our interested time series to the frequency domain. The main blocks of variation in frequency domain are sinusoid functions. A time series $X[n]$, where n is a positive integer and limited to N, can be defined as a linear combination of some sinusoids with random amplitudes and fixed frequencies $\{f_i\}$ [11]:

$$X_t = \mu + \sum_{i=1}^{n/2} [A_i cos(2\pi f_i t) + B_i sin(2\pi f_i t) \qquad t = 1, 2, ..., N \tag{3.1}$$

where μ is a constant, A_i and B_i are the amplitudes of the ith sinusoid component. The frequency f_i is related to N by:

$$f_i = \frac{i}{N} \qquad 1 < i < \frac{N}{2} \tag{3.2}$$

We assume that A_i and B_i are zero-mean independent random variables with the same variance σ_i^2. With these assumptions we come to a conclusion that μ is the expected value of $X[n]$:

$$E\{X_t\} = \mu \tag{3.3}$$

Then the variance of $X[n]$ is:

$$\sigma^2 = E\{(X_t - \mu)^2\} = \sum_{i=1}^{N/2} \sigma_i^2 \tag{3.4}$$

and the autocorrelation function (ACF) of $X[n]$ is:

$$\rho_k = \frac{\sum_{i=1}^{N/2} \sigma_i^2 cos(2\pi f_i k)}{\sum_{i=1}^{N/2} \sigma_i^2} \tag{3.5}$$

Equation (3.4) indicates that the total variance is the sum of variances corresponding to the sinusoidal comportments at the different standard frequencies; therefore, the variance of the time series is expressed as a function of frequency. This function is expressed as the spectrum of $X[n]$. From (3.5) we find a relation between ACF and spectrum. If we consider σ_i^2 as a spectral value, the ACF is a cosine transform of the spectrum, similarly, the spectrum is the Fourier transform of the ACF; so, ACF and Spectrum are complementary to each other, one in time domain and the other in frequency domain [12].

3.1. Spectral Analysis

Spectral analysis is related to estimating the unknown spectrum of a process from a limited number of data and finding the important frequency components contributing to the variance of the signal [12]. In general, there are two main methods to extract the frequency characteristic of a random process: non-parametric and parametric. In non-parametric method the spectrum of the signal is obtained by taking the discrete Fourier transform (DFT) using fast Fourier transform (FFT); on the other hand, in the parametric approach, the signal is modeled as the response of a linear time invariant (LTI) system, fed by a white noise as its input.

3.1.1. Non-parametric Approach

In non-parametric methods, also called DFT-based methods, there is no prior assumption about the model of spectrum. Recall that the spectrum is the Fourier transform of ACF of a time series. So if we estimate the ACF, then we can find spectrum just by taking DFT. That is to say:

$$P(f) = \sum_{k=-\infty}^{k=\infty} r(k) e^{-j2\pi f_k} \tag{3.6}$$

The problem is that, in reality we do not have infinite number of samples of a random process. In practice, we estimate the autocorrelation with the limited number of samples of the observed time series. The ACF can be estimated from observed data $X[0]$, $X[1]$, ..., $X[N]$ by:

$$r(k) = \frac{1}{N} \sum_{n=0}^{N-1-k} X^*[n]X[n+k] \tag{3.7}$$

where $X^*[n]$ is the conjugate of $X[n]$. Having estimated ACF, the spectrum is obtained by using DFT. Generally, the non-parametric methods are considered as multiplying a time series by a window and then estimating the PSD of that specific window [12]. The simplest and most famous DFT-based method which implements rectangular window is Periodogram method [12]. According to the window shape there are variety of non-parametric methods such as Welch, Bartlett and so on [13, 14].

3.1.2. Parametric Approach

The alternative methods of PSD estimation are parametric methods. The assumption is that the random process can be modeled by a parametric model; therefore, "model-based methods" is another name for these approaches. They are further categorized to two different approaches: based on the polynomial model, or stand on the eigenvector decomposition. In the two following sections, we will describe two different model-based approaches of autoregressive and multiple signal classification methods.

3.1.2.1. Polynomial-based Approaches. In these methods, the observed time series is assumed to be generated by processing a white noise ($u[n]$) with power of σ^2 through a rational stable and causal filter with the transfer function $H(f) = \frac{B(f)}{A(f)}$. In the time domain the filter equation is defined as:

$$X[n] + \sum_{k=1}^{p} a_k X[n-k] = \sum_{l=1}^{q} b_l u[n-l] \tag{3.8}$$

The transfer function of this system is represented as:

$$H[f] = \frac{\sum_{l=1}^{q} b_l e^{-j2\pi f_l}}{1 + \sum_{k=1}^{p} a_k e^{-j2\pi f_k}} \tag{3.9}$$

Therefore the spectrum of the output which is our desired spectrum estimation, is obtained by:

$$P(f) = \frac{\left|\sum_{l=1}^{q} b_l e^{-j2\pi f_l}\right|^2 \times \sigma^2}{\left|1 + \sum_{k=1}^{p} a_k e^{-j2\pi f_k}\right|^2} \tag{3.10}$$

If both p and q are non-zero, the system is called autoregressive moving average and is abbreviated by ARMA(p, q). If q is equal to zero and p is non-zero, the system is called autoregressive and is abbreviated by AR(p) and finally, if p is zero and q is non-zero the system is called moving average and it is abbreviated by MA(q). In this thesis, we just concentrate on AR model. In AR, the time series is generated by a causal filter which its input-output difference equation is given by:

$$X[n] = -\sum_{k=1}^{p} a_k X[n-k] + u[n] \tag{3.11}$$

The transfer function of the model in (3.9) is simplified to

$$H[f] = \frac{1}{1 + \sum_{k=1}^{p} a_k e^{-j2\pi f_k}} \tag{3.12}$$

and the corresponding estimated spectrum is derived by:

$$P(f) = \frac{\sigma^2}{\left|1 + \sum_{k=1}^{p} a_k e^{-j2\pi f_k}\right|^2} \tag{3.13}$$

To obtain the spectrum of the time series with autoregressive approach, we must find $\{a_k\}_{k=1}^{p}$ and the variance of the input noise (σ^2). There are several methods to estimate the parameters of AR model. Here we introduce three important ones: autocorrelation (Yule-Walker), covariance, and modified covariance methods [15].

3.1.2.2. Autocorrelation (Yule-Walker) Method. Considering (3.11), the autocorrelation coefficient relation between the input and output is written as:

$$r[n] + \sum_{k=1}^{p} a_k r[n-k] = \sigma^2 \tag{3.14}$$

If we rewrite this equation for $n = 0$ to $n = p$, we obtain a set of linear equations which can be arranged in a matrix form as below:

$$\begin{bmatrix} r(0) & r(-1) & \dots & r(-n) \\ r(1) & r(1) & \dots & r(-n+1) \\ \vdots & \vdots & \ddots & \vdots \\ r(n) & r(n-1) & \dots & r(0) \end{bmatrix} \begin{bmatrix} 1 \\ a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} \sigma^2 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{3.15}$$

Additionally, by using $r(-k) = r^*(k)$ and because the signal is real $r(-k) = r(k)$, we can write

$$\begin{bmatrix} r(0) & r(1) & \dots & r(n) \\ r(1) & r(1) & \dots & r(n-1) \\ \vdots & \vdots & \ddots & \vdots \\ r(n) & r(n-1) & \dots & r(0) \end{bmatrix} \begin{bmatrix} 1 \\ a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} \sigma^2 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{3.16}$$

We call (3.16) as the autocorrelation normal equation. If we know $\{r(k)\}$, then we can use all linear row equations except the first row to find $\{a_i\}$. Having found all coefficients, we substitute them in (3.16) and find the variance of noise (σ^2). The most left $p \times p$ matrix is autocorrelation (R) matrix, which is a Toeplitz matrix. If the R matrix is positive definite, all roots are inside the unit circle and the system is stable. In summary, to estimate the parameters of AR model using Yule-Walker estimation method, first we estimate the autocorrelation coefficients of the observed time series by using (3.14) and then we make use of the set of linear equations in (3.16) to calculate the AR parameters. Because of the window effect caused by estimating autocorrelation function, the Yule-Walker approach usually is not used in the case of short-length data series.

3.1.2.3. Covariance Method. In this method, parameters estimation is accomplished by finding the solution to the set of linear equations:

$$\begin{bmatrix} r(1,1) & r(2,1) & \dots & r(n,1) \\ r(1,2) & r(2,2) & \dots & r(n,2) \\ \vdots & \vdots & \ddots & \vdots \\ r(1,n) & r(2,n) & \dots & r(n,n) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} r(0,1) \\ r(0,2) \\ \vdots \\ r(0,n) \end{bmatrix} \tag{3.17}$$

where

$$r(k,l) = \sum_{m=n}^{N-1} X(m-l)X^*(m-k) \tag{3.18}$$

This matrix is not Toeplitz as the same as that one in autocorrelation method. The advantage of this method over the previous one is that there is no window assumption; therefore, for short data set the resolution of the obtained spectrum by this method is higher than the results of autocorrelation method.

3.1.2.4. Modified Covariance Method. This approach is very similar to the covariance method and the coefficients of AR are calculated by (3.17). The difference is in the way of estimating autocorrelation parameters given by:

$$r(k,l) = \sum_{m=n}^{N-1} [X(m-l)X^*(m-k) + X(m-n+l)X^*(m-n+k) \tag{3.19}$$

which is derived by minimizing the sum of the squares forward and backward errors:

$$\varepsilon_n(m) = \varepsilon_n^+(m) + \varepsilon_n^-(m) = \sum_{m=n}^{N-1} [|e_n^+(m)|^2 + |e_n^-(m)|^2] \tag{3.20}$$

where

$$e_n^+(m) = X(m) + \sum_{k=1}^{n} a_n(k)X(m-k) \tag{3.21}$$

and

$$e_n^-(m) = X(m-n) + \sum_{k=1}^{n} a_n^*(k) X(m-n+k) \tag{3.22}$$

are forward and backward prediction errors respectively [12]. The advantage of this method over two previous ones, is that it gives statistically stable spectrum estimates with higher resolution.

3.1.2.5. Eigenanalysis-based Methods. Eigenanalysis-based methods are those methods which assume that the signal consists of the sum of N sinusoid signals contaminated with white noise. That is to say, we can consider the time series $X[n]$ as:

$$X[n] = \sum_{k=1}^{N} A_k e^{jw_k n} + u[n] \tag{3.23}$$

where $\{A_k\}$ are complex numbers that represent the amplitude and phase of kth exponential components and $u[n]$ is a white noise. The autocorrelation of (3.23) is written as:

$$R_x = R_{signal} + R_{noise} = \sum_{k=1}^{N} |A_k|^2 e_k e_k^{*T} + \sigma_0^2 I \tag{3.24}$$

where $e_k = [1\ e^{jw_k}\ e^{jw_k}\ ...\ e^{j(M-1)w_k}]$ is the kth eigenvectors of R_{signal} matrix. Equation (3.24) can be rewritten as: $R_x = E\Lambda E^{*T} + \sigma_0^2 I$, where $E = [e_1, ..., e_N]_{M\times N}$ and

$$\Lambda = \begin{bmatrix} |A_1|^2 & 0 & \dots & 0 \\ 0 & |A_2|^2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & |A_N|^2 \end{bmatrix}_{M\times N} \tag{3.25}$$

Therefore, the autocorrelation matrix is decomposed into signal and noise subspaces.

3.1.2.6. Pisarenko Harmonic Decomposition. In this approach, we assume that $M = N + 1$, that is to say, the dimension of signal subspace is N and that of the noise is one [16]. There is just one eigenvector V_n and its corresponding eigenvalue $\lambda_n = \sigma_0^2$ for the noise subspace and it is orthogonal to the subspace of the signal. We can write

$$e_i^{*T} u_n[k] e^{-jw_i k} = 0 \tag{3.26}$$

This equation results in a statement called annihilating filter described by:

$$U_n(z) = \sum_{k=0}^{N} u_n[k] z^{-k} = \prod_{k=0}^{N} (1 - e^{jwk} z^{-1}). \tag{3.27}$$

The frequencies of the signal are related to the angular positions of the zeros of this filter in (3.27) which reside inside the unit circle. Assuming all eigenvectors are unit norm, we can write:

$$\begin{gathered} u_i R_x = \lambda_i u_i \\ u_i^{*T} R_x u_i = \lambda_i u_i^{*T} u_i = \lambda_i \\ u_i \left[\textstyle\sum_{k=1}^{N} |A_K|^2 e_k e_k^{*T} + \sigma_0^2 I \right] = \lambda_i \\ \textstyle\sum_{k=1}^{N} |A_k|^2 \left| e_k^{*T} u_k \right|^2 = \lambda_i - \sigma_o^2 \end{gathered} \tag{3.28}$$

Having calculated signal frequencies, we can find the powers $|A|^2$ by (3.28). By evaluating (3.28) at different frequencies it is possible to obtain the so-called pseudo-spectra,

$$\tilde{P}(e^{jw}) = \frac{1}{\left| e^{*T}(w) u_{min} \right|^2} \tag{3.29}$$

3.1.2.7. Multiple Signal Classification (MUSIC). This method was introduced in [17]. To improve the performance of Pisarenko estimator [16], the averaging was proposed in [17]. This method takes advantage of using of multiple noise eigen-filters instead of one which is used in [16]. The MUSIC procedure is based on the assumption that the observed time series consists of a series of N complex sinusoids corrupted by an additive white noise. Instead of having just one annihilating filter, MUSIC uses $M - N$

eigenfilters. First the autocorrelation matrix of the time series R_x is estimated. The eigenvectors $V_1,\ V_2,\ ...,\ V_M$ and corresponding eigenvalues $\lambda_1,\ \lambda_2,\ ...,\ \lambda_M$ are extracted from the matrix. The first N large eigenvectors V_1 to V_N are from signal subspace and the rest of eigenvectors belong to the noise subspace. A group of complex sinusoids $e(w_1), e(w_2), , e(w_N)$ vectors are employed to represent the signal subspace. If the time series has a component at the w_i frequency, the vector $e(w_i)$ is orthogonal to $V_{N+1}, V_{N+2}, , V_M$; therefore, the peak value at w_i is computed by

$$PSD(w_i) = \frac{1}{\sum_{N+1}^{M} |e^T(w_i)v_k|^2} \times \frac{1}{\Delta w} \tag{3.30}$$

It should be noted that this method just indicates the frequency location in the time series; as a result, the magnitude of each peak does not show the PSD at the corresponding frequency; so instead of PSD it is known as the pseudo PSD. The power spectral density obtained by this technique has higher resolution in comparison with those estimated by the FFT-based methods.

3.1.3. Non-parametric vs. Parametric

Because DFT-based methods carry out the estimation by windowing data, they can introduce distortion to the estimated PSDs. The main advantage of these approaches is that the PSDs estimated by them do not have spurious frequency peaks; on the other hand, model-based methods do not apply data windowing. If the PSDs are obtained by wrong models in the model-based methods, they may consist of spurious frequency peaks. By choosing an appropriate model to estimate PSD, the results usually are less biased and have a lower variances than those of estimated by non-parametric methods.

3.1.4. Related Works

Despite the fact that DoS and DDoS attacks are new concepts in network security problems, there are many papers considering the methods of detection and prevention of these types of attacks. In this section we go through some previous works regarding to DDoS attack detection using frequency domain.

Cheng *et al.*, in a study against DoS attack, proposed a method to identify normal traffic to decrease the probability of false positive [18]. They considered TCP traffic as the dominant normal traffic. Making use of periodicity in this type of protocol due to the round time trip (RTT), they suggested that by analysing the PSD, the normal traffic can be detected. They found the PSD using Welch windowing method [13].

In another works done by Alefiya *et al.*, the authors used a combination of three different methods to detect DoS and DDoS attacks [19]. The method consisted of packet-header, transient ramp up and spectral analysis. For the spectral approach they implemented normal DFT method, and to compare the results of abnormal traffics, they employed normalized cumulative spectrum (NCS). The 60% of this value called $F(60\%)$ in an attack used as the threshold to segregate DoS and DDoS. If this value was located in lower frequencies, it was an indication of DDoS attack; otherwise, the attack was considered to be a DoS traffic.

A special type of DDoS attack called shrew attack was considered in a work done by Chen *et al.* [20]. They used the spectrum analysis to distinguish shrew from normal traffic. The probability density functions of NCS of both normal and shrew were estimated. A threshold was defined to separate traffics. If the value of the NCS was less than the threshold, the traffic was considered as normal; otherwise, it was shrew attack.

In a study done by Hashim *et al.*, authors investigated the frequency signatures of DoS and DDoS attacks in the next generation mobile network (NGMN) [21]. They used these aspects as the signature to detect abnormality in the network. The Lomb periodogram algorithm was employed to estimate the PSD of the traffic [22]. According to their results obtained from a simulation, the main energy of DDoS attack resided in lower frequency, however for DoS higher frequencies were dominant. To detect the suspicious activity in the traffic, they proposed two levels of detection. At the first level, called "Mirror" effect, the cross-correlation of the underlying traffic and signature was obtained and if it was symmetric, they considered it as an attack. At the second stage, known as "Derivative", the derivative of the cross-correlation was estimated and according to its value the DoS and DDoS attack were segregated from each other.

Although in all studies, authors investigated the frequency domain to detect denial of service attacks, but non of them considered parametric methods to find the spectrum of the traffic. In addition, most of these studies were done in the simulation environment. They also did not consider a practical model which can be implemented in an IDS system. In this work, we not only consider parametric methods but also, we compare it with a DFT-based approach. The data which we have used come from a experimental result of DDoS attack. We also implement the results in a Signature-based intrusion detection system.

4. SIGNAL DECOMPOSITION BASED ON BASIC MATCHING PURSUIT ALGORITHM

Representing an original signal with its fundamental constituents, which is so called "signal decomposition", is a ubiquitous method in signal processing. Decomposing a signal not only provides a simple signal free from redundancy but also makes sophisticated operations much simpler. In this approach the underlying complicated signal is represented by a combination of its fundamental features called function segments or atoms. Recently, a new method of decomposition called sparse representation has caught the attention of scientists. In this method a signal $y \in \mathbb{R}^n$ is decomposed by using an over-complete dictionary matrix $D \in \mathbb{R}^{n \times k}$ consisting of k signal-atoms for columns $\{d_j\}_{j=1}^{k}$. The model Dx obtained for y must satisfy $||y - Dx|| \leq E$. $x \in \mathbb{R}^k$ comprises of representation coefficients. The process of estimating these coefficients is called "sparse coding " or "atom decomposition" and it is usually accomplished by pursuit algorithms. In the following section we will describe one of the popular feature extraction algorithms called basic matching pursuit decomposition.

4.1. Classical Matching Pursuit Decomposition

The matching pursuit decomposition (MPD) was originally introduced by Mallat and Zhang [23]. MPD is an iterative algorithm. In each stage, the best atom is determined by matching a portion of the signal to the segment functions. Once the best match is obtained, it is extracted from signal and the algorithm proceeds onto the next iteration. As a greedy based algorithm, MPD obtains the maximum amount of energy possible per iteration [3]. The degree of similarity between atoms and the signal is accomplished by cross-correlation. In each iteration, the best match is that one with the largest cross-correlation value and its corresponding time delay. By definition $R_x^k[n]$ is the residual after k iterations, where the signal is $x[n]$. Prior to the first iteration an initial value must be assigned to the residual which is shown in (4.1). The best fit atom is chosen from dictionary D via performing cross-correlation with the residual shown in (4.2). The nominated atom is subtracted from the signal (4.1). The next step starts with the new residual to find the next atom. $d_{\gamma i}$ represents the ith matched atom and a_γ is it corresponding cross-correlation.

$$R_x^0[n] = x[n] \tag{4.1}$$

$$a_\gamma = argmax_{d_{\gamma i} \in D} \left| \left\langle R_x^k[n], d_{\gamma i} \right\rangle \right| \tag{4.2}$$

$$R_x^k[n] = R_x^{k-1}[n] - \left\langle R_x^k[n], d_{\gamma i} \right\rangle a_\gamma \tag{4.3}$$

To avoid of modeling noise or spurious features, if one of the following three criteria is met the process will be ceased: a specified number of atoms are selected, the specified amount of energy is extracted, or the d components remove below a certain threshold amount of energy. Figure 4.1 displays the pseudo code of the MPD algorithm. The process starts by generating a dictionary, and continues extracting the features until one criterion is met. The original signal is reconstructed as :

```
Build dictionary: $D = \{d_{\gamma 1},\ d_{\gamma 2},\ ...,\ d_{\gamma n}\}$, where $d_{\gamma n} = \frac{1}{\sqrt{\alpha_n}} d\left(\frac{t-\beta_n}{\alpha_n}\right) e^{j2\pi k_n t}$.
Initialize $K_{stop}$, $\delta_{stop}$, $K = 0$, $R_x^0[n]$, $E_x^0 = ||R_x^0||_2$.
**MPD routine**.
**while** $k \leq K_{stop}$ $or$ $E_x^k \geq \delta_{stop}$ **do**
  $\alpha_{\gamma j}^k = \langle R_x^k[n], d_{\gamma j}^k \rangle$.
  Select dictionary element whose time correlation with the $R_x^k[n]$ is maximum.
  $R_x^k[n] = R_x^{k-1}[n] - \alpha_{\gamma j}^k d_{\gamma j}^k[n]$.
  $k = k + 1,\ E_x^k = ||R_x^k||_2$
**end while**
```

Figure 4.1. Matching pursuit algorithm. (reprinted from [3]).

$$\hat{x} = \sum_{i=1}^{N} a_{\gamma i} d_{\gamma i} \tag{4.4}$$

4.2. K-SVD Dictionary Generation Method

K-SVD method was introduced by Aharon *et al.* [4]. This approach finds the best dictionary from an over-complete dictionary for sparse representation of a signal. A set of training signal $Y = \{y_i\}_N \in \mathbb{R}^n$ is fed to the system and the best fit dictionary is obtained. The goal of this method is:

$$min_{D,Z} \left\{ ||Y - DZ||_F^2 \right\} \quad subject\ to\ \ \forall_i, ||z_i||_0 \leq T_0 \tag{4.5}$$

Dictionary searching is achieved in two steps: sparse coding and codebook update stages. At the first stage, the dictionary D is assumed to be constant and the sparse coding process is carried out for N members of Y to find the corresponding matrix Z. The penalty is written as:

$$||Y - DZ||_F^2 = \sum_{i=1}^{N} ||y_i - Dz_i||_2^2 . \tag{4.6}$$

By substituting (4.6) in (4.5) we obtain:

$$min_{z_i} \left\{ ||y_i - Dz_i||_2^2 \right\} \quad subject\ to\ \ ||z_i||_0 \leq T_0 \quad i = 1, 2, ..., N \tag{4.7}$$

The problem (4.7) is addressed by using matching pursuit algorithm discussed in previous section. In the second step, one column of dictionary D, d_k and its corresponding coefficients residing in the *ith* row of Z, denoted as Z_T^i are updated. The equation (4.5) is written as:

$$||Y - DZ||_F^2 = \left|\left| Y - \sum_{j=1}^{K} d_j z_T^j \right|\right|_F^2 = \left|\left| \left(Y - \sum_{j \neq k} d_j z_T^j \right) - d_k z_T^k \right|\right|_F^2 = \left|\left| E_k - d_k z_T^k \right|\right|_F^2 \tag{4.8}$$

E_k is the error of estimation when the kth atom is excluded. A group of y_is, those use the atom d_k, is defined as:

$$w_k = \left\{ i | 1 \leq i \leq K, z_T^k(i) \neq 0 \right\} \tag{4.9}$$

Return to (4.5), the minimization is now equivalent to minimize:

$$\left|\left| E_k^R - d_k z_R^K \right|\right|_F^2 \tag{4.10}$$

where Z_R^k is the minimized version of Z_T^k considering only non-zero entries, and E_K^R is obtained from E_k by choosing columns corresponding to w_k. E_K^R is decomposed using SVD decomposition method [24] to $E_K^R = U \Delta V^T$. The first column of U is the update $\hat{d}_k$, and the product of the first column of V and $\Delta(1,1)$ gives the coefficients of Z_R^K. Figure 4.2 displays the psudo code of K-SVD method.

```
Task: Find the best dictionary to represent the data samples $\{y_i\}_{i=1}^{N}$ as sparse compositions, by solving
    $$min_{D,Z}\{||Y - DZ||_F^2\} \;\; subject\ to\ \forall i,\ ||z_i|| \leq T_0.$$
Initialization: Set the dictionary matrix $D^{(0)} \in \mathbb{R}^{n \times k}$ with $l^2$ normalized columns. Set $J = 1$.
while Convergence does not meet do
  Sparse Coding Stage: Use any pursuit algorithm to compute the representation vectors $z_i$ for each example $y_i$, by approximating the solution of
      $$i = 1,\ 2,\ ...,\ N, \quad min_{z_i}\{||y_i - Dz_i||_2^2\} \;\; subject\ to\ ||z_i||_0 \leq T_0.$$
  Codebook Update stage:
  for all $k = 1,\ 2,\ ...,\ K$ in $D^{J-1}$ do
    Define the group of examples that use this atom, $w_k = \{i|1 \leq i \leq N,\ x_T^k(i) \neq 0\}$.
    Compute the overall representation error matrix, $E_k$, by
        $$E_k = Y - \sum_{j \neq k} d_j x_T^j.$$
    Restrict $E_k$ by choosing only the columns corresponding to $w_k$, and obtain $E_k^R$.
    Apply SVD decomposition $E_k^R = U \Delta V^T$. Choose the updated dictionary column $\hat{d_k}$ to be the first columns of U. Update the coefficient vector $x_R^k$ to be the first column of V multiplied by $\Delta(1,\ 1)$.
  end for
  Set $J = J + 1$.
end while
```

Figure 4.2. K-SVD Algorithm. (reprinted from [4]).

5. EXPERIMENTAL REVIEW

In this section, we will discuss experiments and results relating to the frequency characteristics of normal and DDoS traffics. This part consists of two subsections: first we obtain the power spectral density of normal traffic and DDoS attack by means of DFT-based and model-based approaches. The features extracted from the first part will be used in a signature-based IDS. The normal and the attack features are used as training and testing data sets for the system respectively.

5.1. Traffic Dataset

In order to gather traffic data, a wired local network consisting of 22 computers, a server, and a router, was created in the Electrical and Electronics Dept. of Boğaziçi University. One computer with Ubuntu 11.10 operated as a server and provided service to its clients. The rest of computers acted as legitimate clients or zombies. Figure 5.1 displays the network layout of this experiment.

To run the normal traffic in the network, we used file transfer protocol (FTP). FTP is a TCP-based network protocol providing the facility of exchanging files through a network. During normal traffic there was no attack in the network. The second traffic pattern was DDoS traffic which was generated by traffic generator software (Tfgen) [25]. During the attack session, all 22 zombies contributed in flooding the victim. To run the attack, we implemented UDP (connectionless computer networking protocol) flood attack. In a UDP flood attack large number of UDP packets are sent to the random ports of the victim. The system is forced into sending back many ICMP packets and finally the system becomes out of reach for its normal users. We used offline process to find frequency features. It means that the data was sampled and then analyzed. Using 'tcpdump' function , a built-in code in Linux-based systems to monitor the network traffic, the total traffics between the victim's node and the rest of the network in both normal and attack sessions were saved for further process. These two tcpdump files were changed into comma separated values (CSV) files by Wireshark software [26].

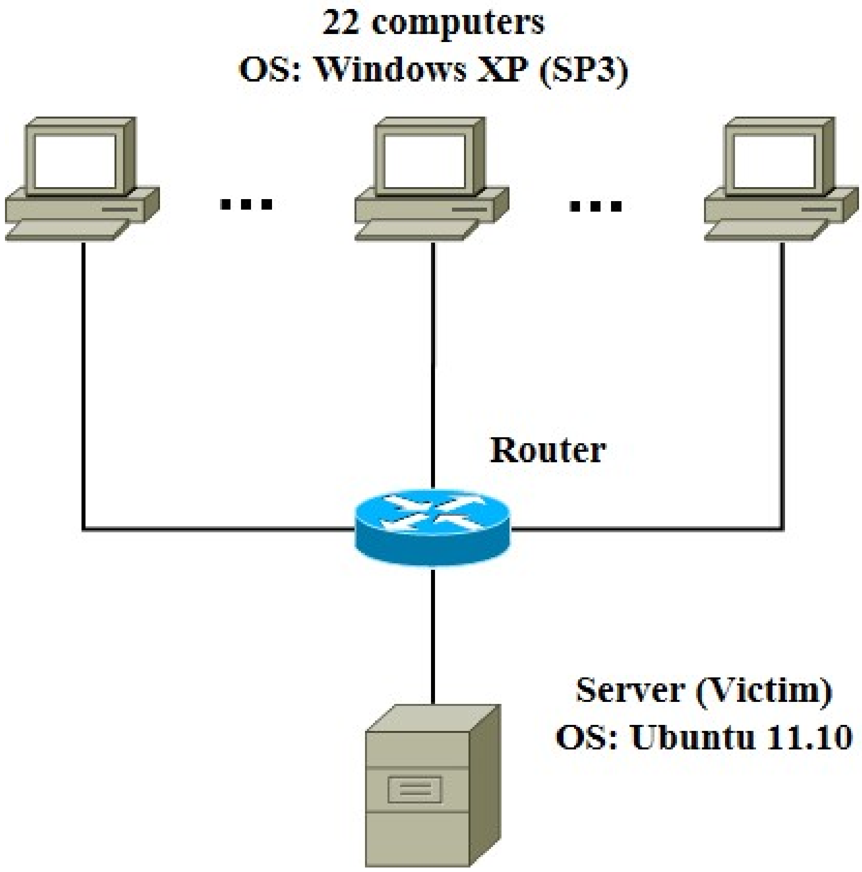


Figure 5.1. Network layout.

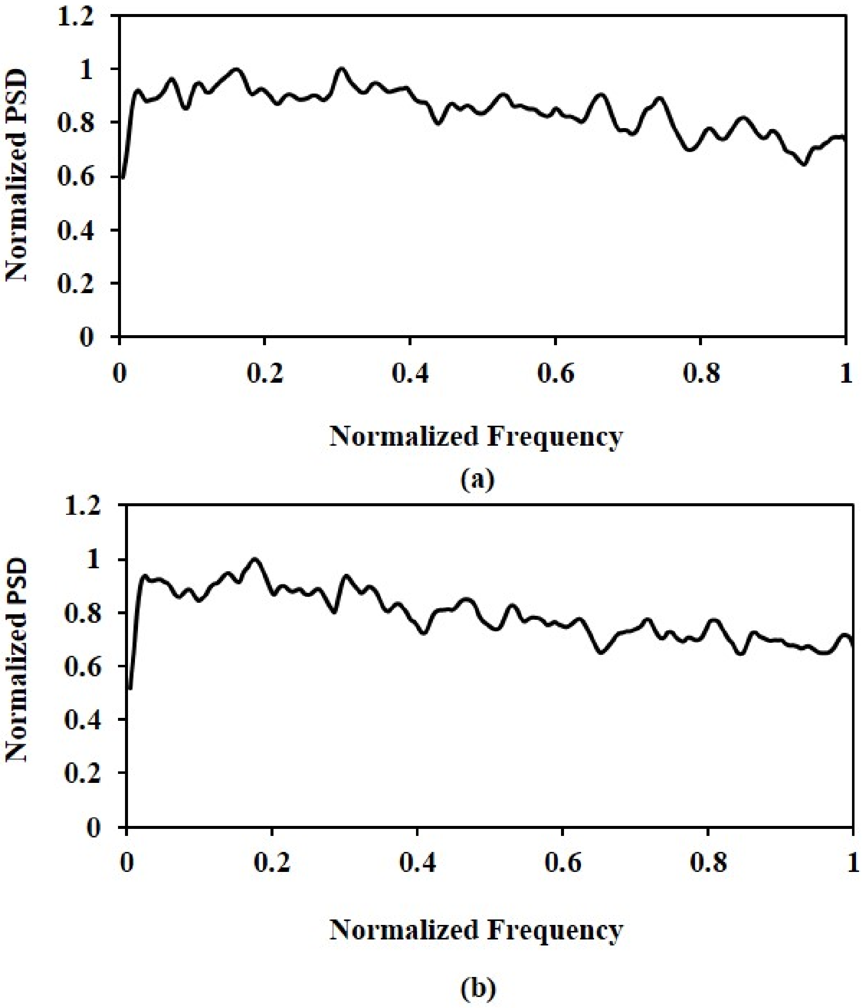
MATLAB R2013b environment was selected to analyze traffic patterns [27].

5.2. Power Spectral Density

For each data set, the number of packets in a specific interval were acquired. These numbers can be modeled as a random process $X[t]$ where $t \in [0\ n]$ and n $\in \mathbb{N}$. The obtained random process was further divided into 100, 128-length subsets. For each subset, the power spectral density was estimated and finally the averaged PSD was calculated.

5.2.1. DFT-based PSD Estimation

For non-parametric method, the Hanning windowing approach was chosen to find the PSD of the traffic [28]. Two different sampling rates of 0.5 ms and 1 ms were used to compare the resolution of the PSDs. According to the Nyquist Sampling theorem the maximum frequency that can be realized with 1 KHz and 2 KHz are 500 Hz and 1 KHz respectively [28]. Figure 5.2 displays the normalized averaged PSD estimated for normal traffic. The frequency axis is normalized to reside between 0 and 1. The energy of normal traffic is almost evenly distributed in different frequencies. Figure 5.3 displays the power spectral density of DDoS attack. In contrast to the spectrum of normal traffic, the main part of energy resides in lower frequencies. Although there are some spikes in higher frequencies, they are negligible in comparison to the amount of energy located in lower part. Considering the PSD distributions of two traffics, the lower bound of frequencies is the distinguishable part which can be used to separate and to detect DDoS attack from the normal one. Furthermore, because we just concentrate



(a)

(b)

Figure 5.2. Normal PSD estimated by DFT method, sampling Frequency, (a) 1 KHz, (b) 2 KHz.

on lower bound of the frequencies, 1 *ms* sampling rate is enough for the detection process.

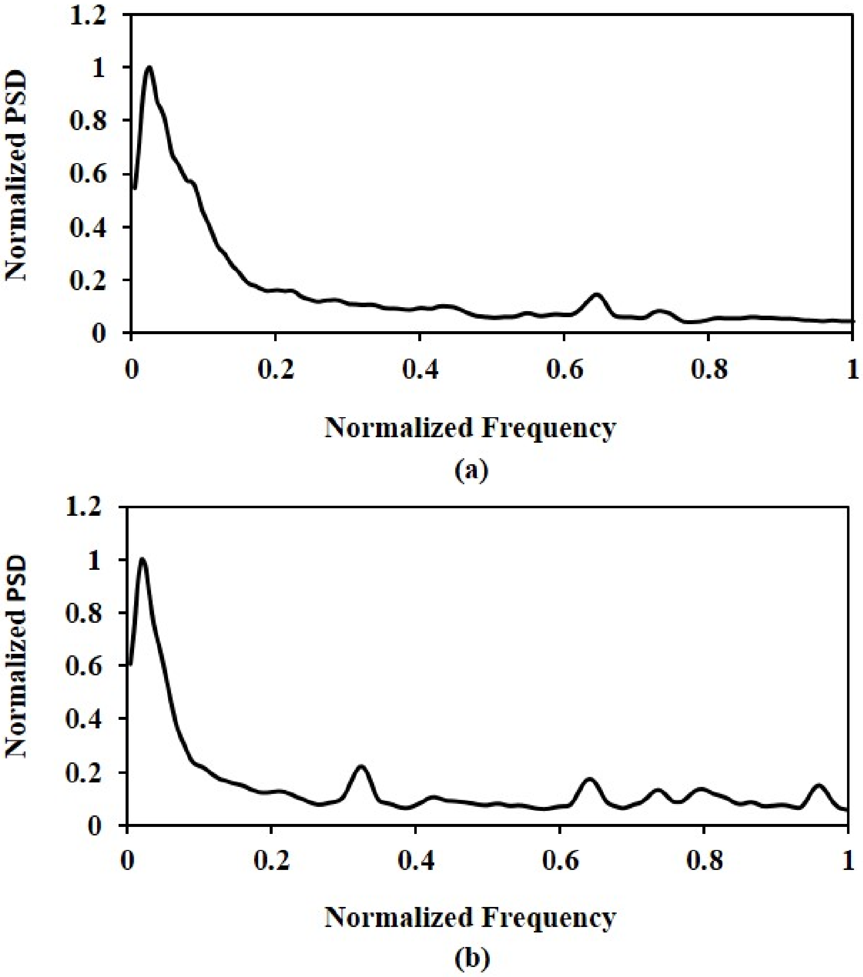


Figure 5.3. DDoS PSD estimated by DFT method, Sampling Frequency, (a) 1 KHz, (b) 2 KHz.

5.2.2. Model-based PSD Estimation

As mentioned before, in contrast to the non-parametric method which is based on the discrete Fourier transform, in the parametric approach, the signal is modeled as the output of a filter whose input is a white noise. In the section we first compare PSDs obtained by AR and MUSIC methods, then we concentrate on the AR and the roots of the polynomial and finally we will discuss the PSDs acquired by AR model.

5.2.2.1. MUSIC vs. AR. The spectrum of normal and DDoS traffics were estimated by two different methods of MUSIC and AR, the former is eigenbased and the latter is polynomial based models. The order was chosen 10 for both methods. MUSIC approach is based on eigenvector decomposition approaches to estimate power spectral density. Because the amplitude of peaks in the estimation carries no information regarding to the true power of each frequency component, the PSD estimated by this method is usually called pseudo PSD. Modified covariance was used to estimate the parameters of AR model. Figure 5.4 compares the results of PSD estimation with these two methods. Figure 5.4(a) makes a comparison between the normal PSDs. The AR result is much smoother than that of MUSIC, and the resolution of MUSIC estimation is higher beside to that of AR method. Figure 5.4(b) compares the PSDs of the DDoS attack, and there is no significant difference between them. In this thesis we just consider AR model, and the MUSIC estimation will be postponed to future works. We continue the rest of this thesis concentrating on AR model.

5.2.2.2. AR Parameter Estimation . The parameters of AR were estimated by three different methods of Yule-Walker, covariance and modified covariance. Figure 5.5(a) displays the average of normalized power spectral density of normal traffic over 100 intervals derived by using these three approaches. The frequency domain of $[0 \ \pi]$ has been normalized to lie between 0 and 1. Despite the fact that Yule-Walker introduces windows effect in estimating PSD, the difference is trivial and all three graphs almost overlap.

Figure 5.5(b) illustrates the results of power spectral density in the case of DDoS attack obtained by three different approaches of solving AR model. The results of covariance and modified covariance overlap each other, and the difference of the Yule-Walkers is also negligible next to two other methods. we can draw a conclusion that all three methods are acceptable enough to estimate power spectral density for both cases. In this experiment we used modified covariance method.

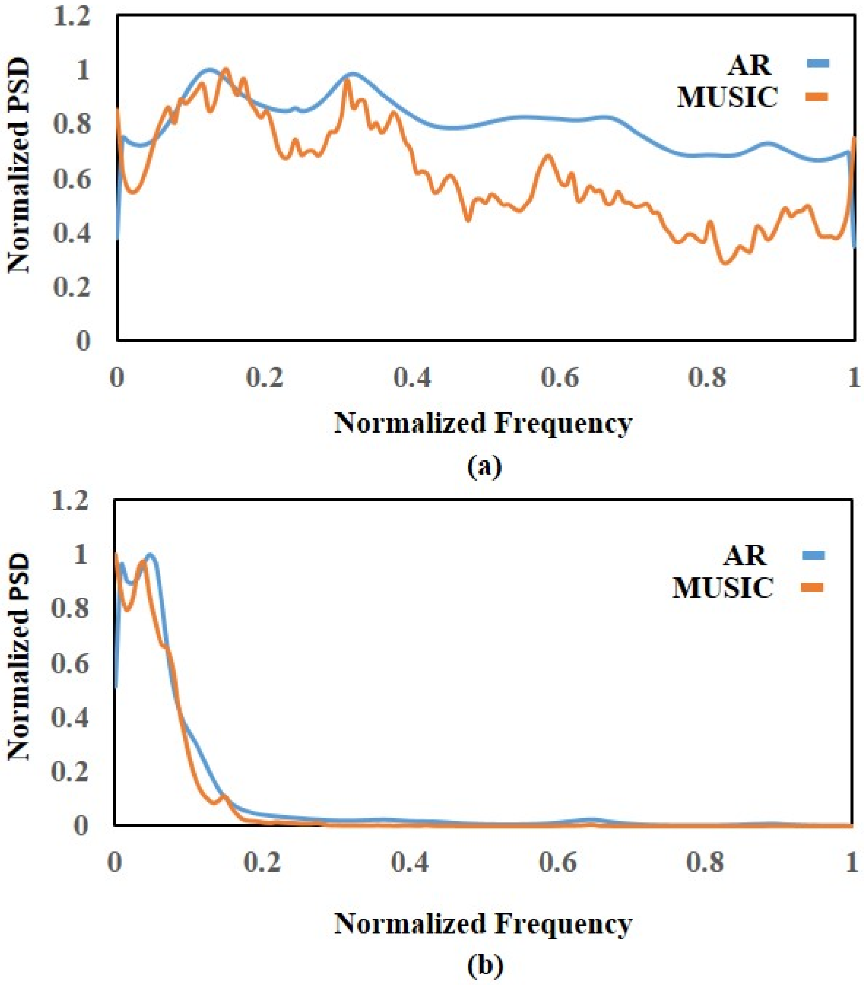


Figure 5.4. PSD estimated by AR and MUSIC, (a) Normal, (b) DDoS.

5.2.2.3. AR Model Order Selection. In employing the AR model, an important consideration which must be taken into account prior to implementation is the order, p, of the model. Small values of p result in a smooth PSD which has low resolution and detail relating to the interested spectrum (underestimating); on the other hand, choosing large values of p introduce feigned details to the obtained PSD (overestimating). In general there are four different methods to estimate the order of the AR including final prediction error (FPE), minimum information theoretical criteria (AIC), criterion autoregressive transfer function (CAT), and first zero crossing (FZC) [29]. Overestimating the order of the model has less effect on the estimation than the underestimating the order. As a rule of thumb, Haykin and Kesler suggest a range between 5 to 20 percent of N as the order of the AR model which is for our experiment ($N = 128$), it is between 6.4 and 25.6 [30]. Ulrych and Ooe recommend that the order lies between

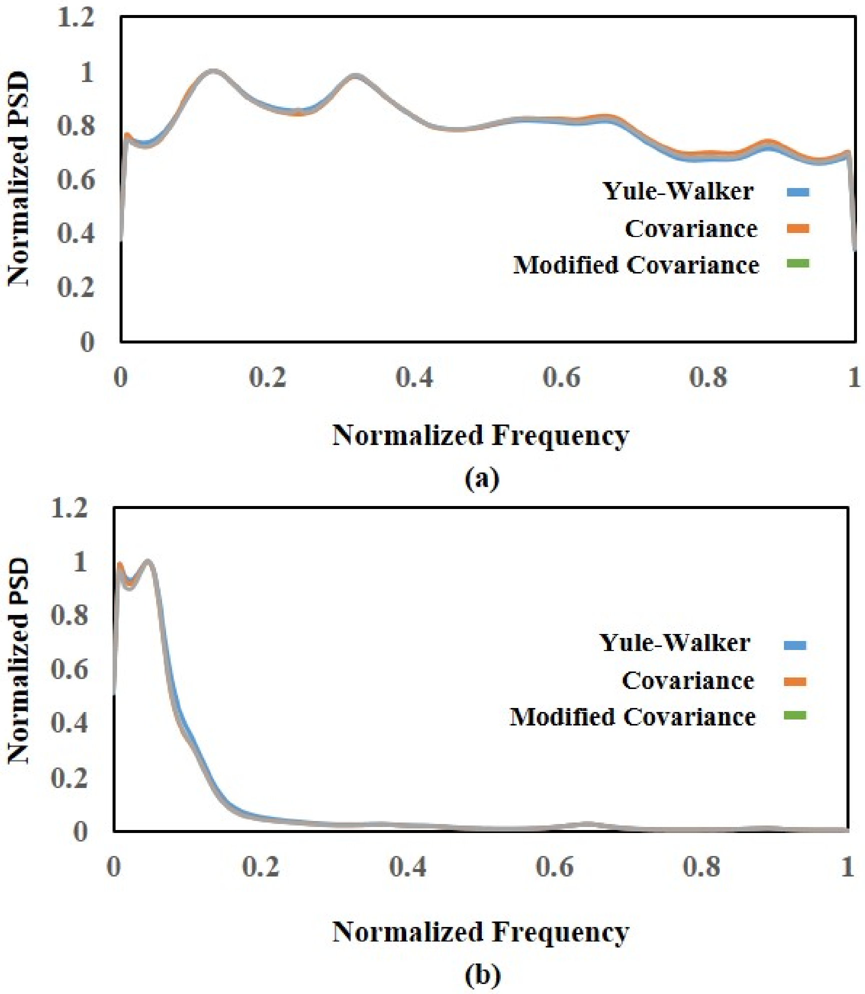
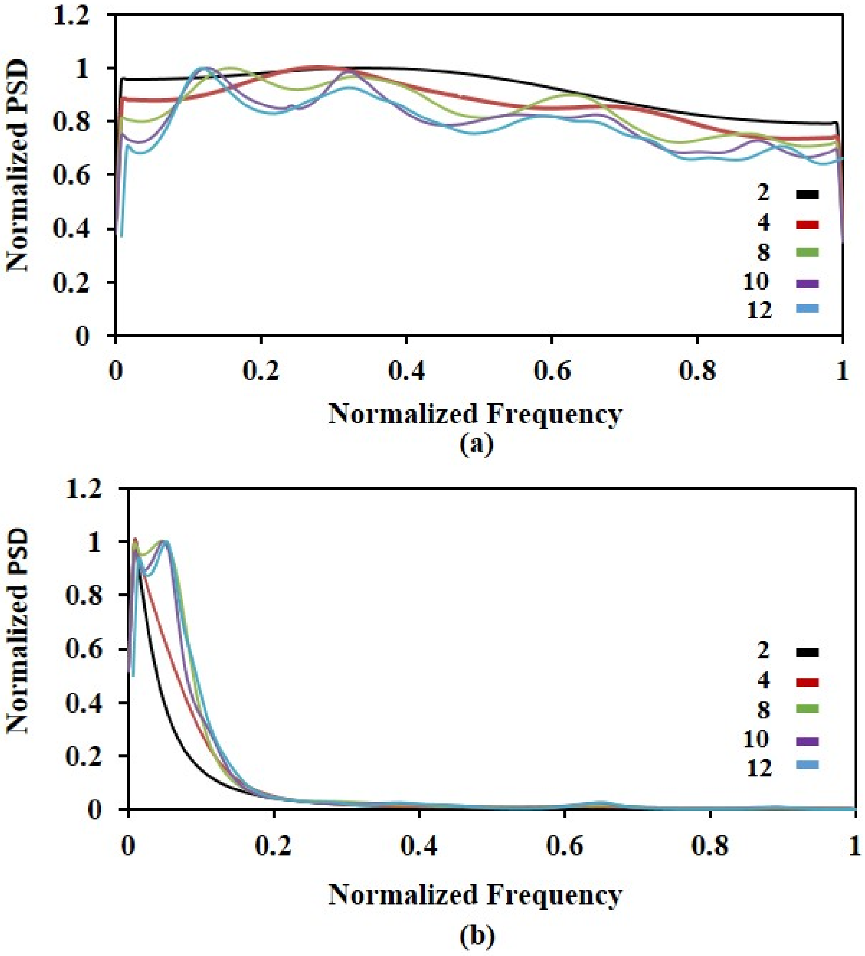


Figure 5.5. PSD estimated by AR using Yule-Walker, covariance, and modified covariance, (a) Normal, (b) DDoS.

$\frac{N}{3} - 1$ and $\frac{N}{2} - 1$ [30]. To investigate the effect of the order of the model on the underlying spectrum, the power spectral density was estimated with different values of p comprising of 2, 4, 8, 10, and 12. Figure 5.6 displays the averaged normalized PSD over 100, 128-length samples of normal and DDoS traffics. The frequency axis is normalized to reside between 0 and 1. For orders of 2 and 4 results are almost smooth and there is no more information regarding to the frequency characteristics of the traffics. For the other orders outcomes are very similar and the difference almost immaterial.

The next important parameter that must be taken into account in AR model selection is the stability of the system. To assure that an AR system is dynamically stable, all roots of so-called characteristic equation must reside inside the unit circle



(a)

(b)

Figure 5.6. PSD estimated by AR using different orders of 2, 4, 8, 10, and 12, (a) Normal, (b) DDoS.

[30]. Figure 5.7 displays the roots distribution of normal and DDoS attack for different orders of 8, 10, 12 (we did not consider orders of 2 and 4 for our estimation regarding to the results obtained from the previous part). The AR model is stable for all different orders, although just in one sample of $12th$ order, two roots are outside the unit circle.

To guarantee to get higher resolution and to assure the stability of the system, the order of the AR model, employed in PSD estimation, was chosen to be 10 in this experiment. Figure 5.8 displays the averaged normalized power spectral density of the normal traffic. The frequency was normalized to locate between 0 and 1. Similar to the DFT-based result, the energy is distributed along different frequencies. Figure 5.9

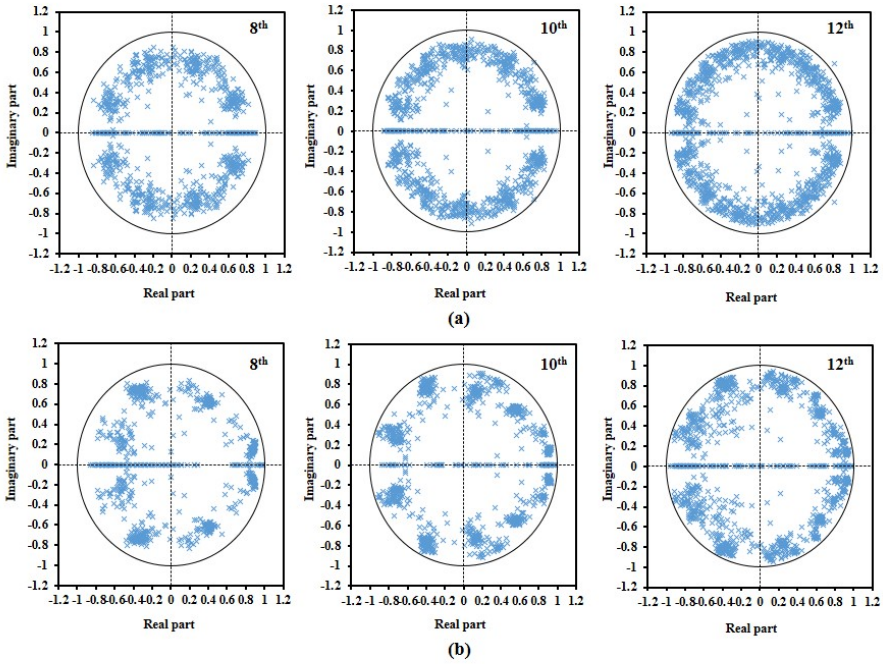


Figure 5.7. Characteristic equation roots distribution of orders : 8, 10, and 12, (a) Normal, (b) DDoS.

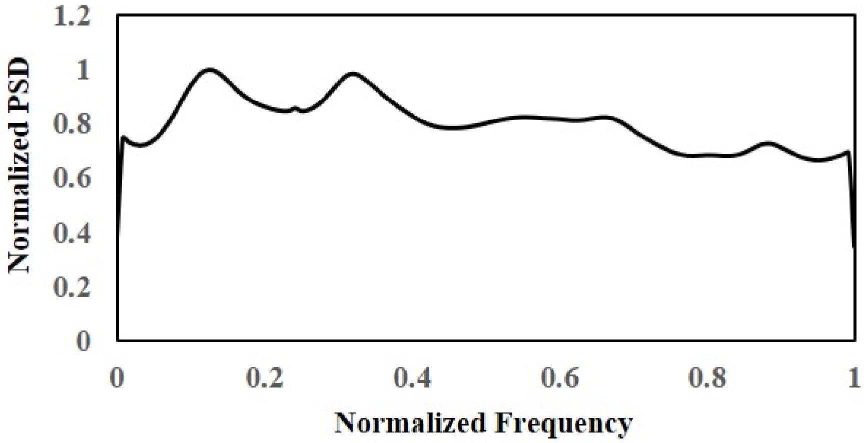


Figure 5.8. Normal PSD estimated by AR(10).

illustrates the averaged normalized spectrum of DDoS attack. Most of the energy is suited in lower bound of frequencies similar to that of DFT-based method.

5.3. Signature-based IDS Using K-SVD Dictionary Generation Method

In the previous section we got two sets of PSDs for each method of spectrum estimation. The first one belongs to the normal traffic and the second one of the DDoS

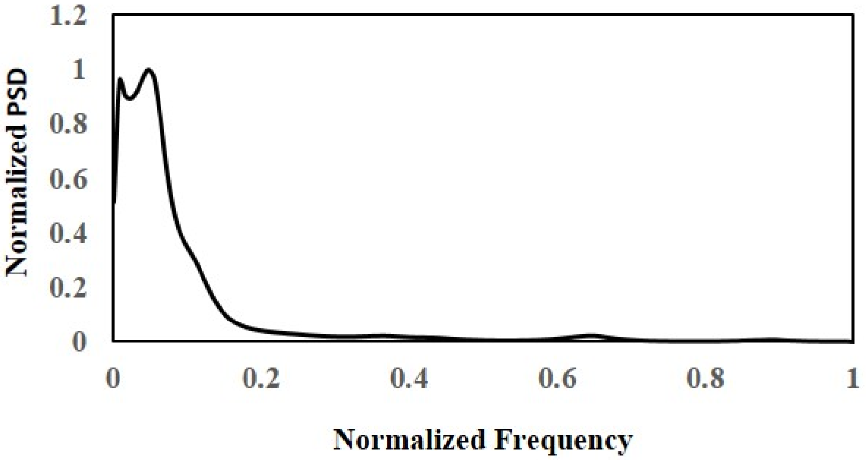


Figure 5.9. DDoS PSD estimated by AR(10).

attack. Each set is a matrix consisting of 100, 128-length vectors of spectrum. In this section we use these to matrices as a training and test signal of a signature-based intrusion detection system illustrated in Figure 5.10. At the first stage, The matrix of normal PSDs is fed to the system as a training data. The system makes use of K-SVD dictionary generator to create the normal signature dictionary with the size of 128×40. The initial matrix is created with the first 40 signals of the training matrix. The maximum number of iterations for the K-SVD algorithm is set to be 30. Figure 5.11 displays the dictionary error norm in each step. At the second stage, the DDoS spectrum set is fed to the system, and it tries to generate signals of the DDoS set by means of the dictionary obtained from the first stage and by applying basic matching

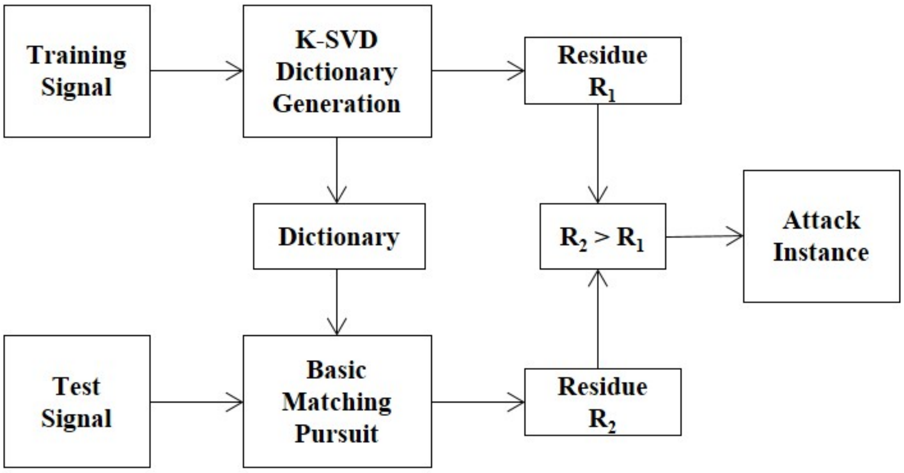


Figure 5.10. Signature-based IDS system block diagram.

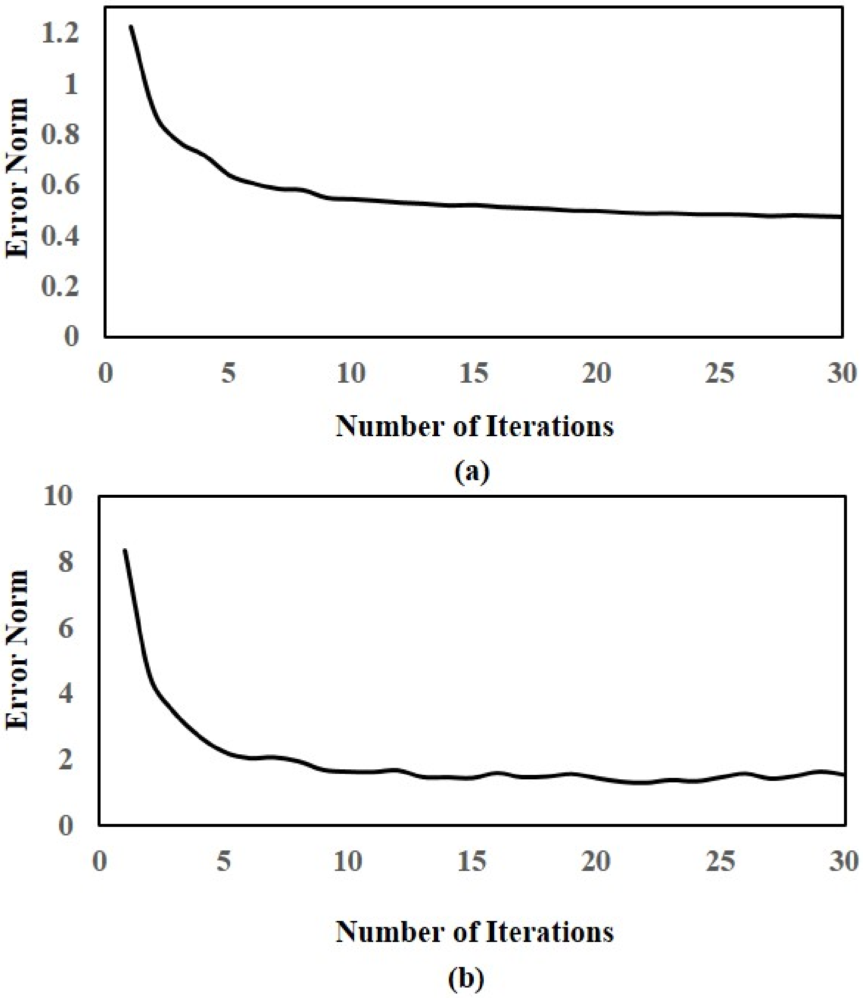


Figure 5.11. Dictionary error norm at each iteration step.

pursuit algorithm. The error norm R_2, the difference between the produced one and the original data, is obtained and compared with R_1. Figure 5.12 displays the receiver operating characteristics (ROC) for these sets of training and test data. For DFT-Based method the ROC is 100%. The threshold of R_1 for DFT-based and Model-based are 1 and 2.1 respectively. The inefficiency of AR approach in modeling the random process or the inappropriate order selection are two possible reasons of low performance of model-based approach regarding ROC.

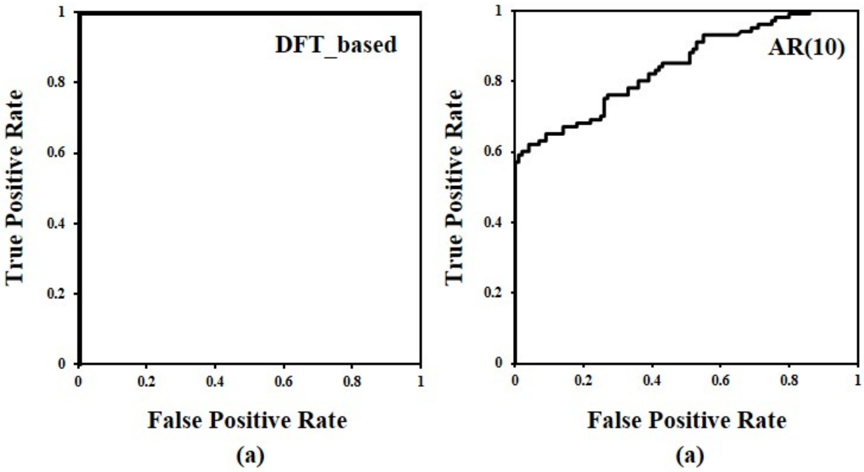


Figure 5.12. ROC, (a) DFT-based, (b) Model-based.

6. CONCLUSIONS

Making a server unavailable for its legitimate users is the main goal of a denial of service attack. In a distributed version of these type of attacks an army consisting of numerous numbers of zombies invades to a victim. The victim begins responding to the meaningless requests from those zombies. Trying to meet the demands from zombies causes the system to not be able to give services to its legitimate users.

One of the approaches to handle DDoS attacks might be increasing the available resource of the victim, but when it comes to a large amount of request it will be also incompetent. When a DDoS attack is running on a system, the first stage is the detection of the source of the attack. A server has many users from different resource so finding the abnormal packets and their sources is necessary. Packet level analysis method is one of the methods of the detection. In this method the content of the packet is examined to find abnormality. Different protocols and applications use some specific packets information that unique for themselves. For example TCP protocol has some specifications which are represented by some bits in the payload of the packets. So if a TCP packet has data other than its usual condition, it may be an attack packet. Attackers disguise their malicious packets by modifying them to be similar to normal ones, and therefore the packet level is impotent to identify attack. Moreover, the IP address in an attack packet is spoofed to make the trace back analysis hard. An alternative method to packet level which can solve the drawbacks of the previous method is flow level analysis. Instead of looking at packets content, the flows of packets are taken into account.

In this thesis, a flow level approach which considering the spectrum characteristics of the packet traffic was employed. At the first step, a test bed consisting of 22 computers, a router and a server (victim) was implemented. Two different traffics including normal and DDoS attack were run in the test bed separately. FTP protocol was used for normal traffic and the DDoS attack was generated by Tfgen software. For each traffic, the number of packets arriving to the node of the server was sampled every 1 ms. The obtained set created a random process that was further divided into 100 128-length subsets.

The frequency characteristic for each subset was estimated with two methods of DFT-based and model-based spectrum estimation. Hanning windowing method was chosen in DFT-based method. For model based approach, AR model with the order of 10 was applied to estimate PSD of the traffic. Finally for each method, two matrices with the size of 100×128 each consisting of PSDs of subsets of normal and DDoS traffics were acquired. According to the results of these two methods of PSD estimation, the energy of normal traffic was distributed in different frequencies and there was no dominant frequency interval; on the other hand, DDoS attack energy mostly resided in lower frequency bound.

At the second step, a signature-based IDS whose dictionary was generated by K-SVD method is implemented. For each method of PSD estimation, the normal matrix was used to train the system and the DDoS one was used to test the system and error residual was obtained. The ROC was sketched for both methods to find the threshold of residual to segregate attack from normal traffic. The performance of the DFT-based method considering the ROC which was 100%, was better than that of the model-based method. The thresholds were 1 and 2.1 for the DFT-based and the model-based respectively. The inefficiency of AR approach in modeling the random process or the inappropriate order selection are two possible reasons of low performance of model-based approach regarding ROC.

In the future work, we will consider other methods of model-based approach rather than AR model to estimate the PSD. Additionally we will consider an anomaly-based IDS instead of the signature-based system.

REFERENCES

1. Ramanauskaite, S. and A. Cenys, “Taxonomy of DoS Attacks And Their Countermeasures”, *Central European Journal of Computer Science*, Vol. 1, No. 3, pp. 355–366, 2011.

2. Specht, S. M. and R. B. Lee, “Distributed Denial of Service: Taxonomies of Attacks, Tools, And Countermeasures.”, *ISCA PDCS*, pp. 543–550, 2004.

3. Christensen, D., S. Das and A. N. Srivastava, “Highly Scalable Matching Pursuit Signal Decomposition Algorithm”, *Proceeding International Workshop on Structural Health Monitoring (IWSHM)*, pp. 1194–1201, 2009.

4. Aharon, M., M. Elad and A. Bruckstein, “K-SVD: Design of Dictionaries For Sparse Representation”, *IN: PROCEEDINGS OF SPARS05*, pp. 9–12, 2005.

5. Tao, Y. and S. Yu, “DDoS Attack Detection at Local Area Networks Using Information Theoretical Metrics”, *Trust, Security and Privacy in Computing and Communications (TrustCom), 2013 12th IEEE International Conference on*, pp. 233–240, IEEE, 2013.

6. Srivastava, A., B. Gupta, A. Tyagi, A. Sharma and A. Mishra, “A Recent Survey on DDoS Attacks and Defense Mechanisms”, *Advances in Parallel Distributed Computing*, pp. 570–580, Springer, 2011.

7. Prasad, K., A. Reddy and K. Rao, “Discriminating DDoS Attack Traffic From Flash Crowds on Internet Threat Monitors (ITM) Using Entropy Variations”, *African Journal of Computing & ICT*, Vol. 6, No. 2, 2013.

8. Van Loon, R. and J. Lo, “An IRC Tutorial”, *Internet Relay Chat (IRC) Help*, 1997.

9. Maheshwari, R., P. Scholar and C. R. Krishna, “Mitigation of DDoS Attacks Us-

ing Probability Based Distributed Hop Count Filtering And Round Trip Time", *International Journal of Engineering*, Vol. 2, No. 7, 2013.

10. Rao, C. and M. W. Ahmed, "Multilayered Probabilistic Model For Distributed DoS Attacks", *Computational Science and Its Applications (ICCSA), 2013 13th International Conference on*, pp. 99–104, IEEE, 2013.

11. Anderson, T. W., *The Statistical Analysis of Time Series*, Vol. 19, John Wiley & Sons, 2011.

12. Box, G. E., G. M. Jenkins and G. C. Reinsel, *Time Series Analysis: Forecasting And Control*, Wiley. com, 2013.

13. Welch, P., "The Use of Fast Fourier Transform For The Estimation of Power Spectra: A Method Based on Time Averaging Over Short, Modified Periodograms", *Audio and Electroacoustics, IEEE Transactions on*, Vol. 15, No. 2, pp. 70–73, 1967.

14. Bartlett, M., "The Spectral Analysis of Point Processes", *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 264–296, 1963.

15. Hayes, M. H., *Statistical Digital Signal Processing And Modeling*, Wiley. com, 2009.

16. Pisarenko, V., "The Retrieval of Harmonics by Linear Prediction", *Geophys. J. R. Astron.*, Vol. 33, pp. 347–366, 1973.

17. Schmidt, R., "Multiple Emitter Location And Signal Parameter Estimation", *Antennas and Propagation, IEEE Transactions on*, Vol. 34, No. 3, pp. 276–280, Mar. 1986.

18. Cheng, C.-M., H. Kung and K.-S. Tan, "Use of Spectral Analysis in Defense Against DoS Attacks", *Global Telecommunications Conference, 2002. GLOBECOM'02. IEEE*, Vol. 3, pp. 2143–2148, IEEE, 2002.

19. Hussain, A., J. Heidemann and C. Papadopoulos, “A Framework For Classifying Denial of Service Attacks”, *Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, SIGCOMM ’03, pp. 99–110, ACM, New York, NY, USA, 2003.

20. Chen, Y., K. Hwang and Y.-K. Kwok, “Filtering of Shrew DDoS Attacks in Frequency Domain”, *Local Computer Networks, 2005. 30th Anniversary. The IEEE Conference on*, pp. 8–pp, IEEE, 2005.

21. Hashim, F., M. R. Kibria and A. Jamalipour, “Detection of DoS and DDoS Attacks in NGMN Using Frequency Domain Analysis”, *Communications, 2008. APCC 2008. 14th Asia-Pacific Conference on*, pp. 1–5, IEEE, 2008.

22. Lomb, N. R., “Least-Squares Frequency Analysis of Unequally Spaced Data”, *Astrophysics and space science*, Vol. 39, No. 2, pp. 447–462, 1976.

23. Mallat, S. and Z. Zhang, “Matching Pursuits With Time-frequency Dictionaries”, *Transaction Signal Processing*, Vol. 41, No. 12, pp. 3397–3415, Dec. 1993.

24. Golub, G. H. and C. Reinsch, “Singular Value Decomposition and Least Squares Solutions”, *Numerische Mathematik*, Vol. 14, No. 5, pp. 403–420, 1970.

25. McKenney, P. E., D. Y. Lee and B. A. Denny, “Traffic Generator Software Release Notes”, *SRI International and USC/ISI Postel Center for Experimental Networking*, 2002.

26. Shimonski, R., *The Wireshark Field Guide: Analyzing And Troubleshooting Network Traffic*, Access Online via Elsevier, 2013.

27. Leang, K., “Matlab Tricks And Tips [Focus on Education]”, *Control Systems, IEEE*, Vol. 33, No. 4, pp. 39–40, 2013.

28. Oppenheim, A. V., R. W. Schafer, J. R. Buck *et al.*, *Discrete-Time Signal Processing*, Vol. 5, Prentice Hall Upper Saddle River, 1999.

29. Schlindwein, F. S. and D. H. Evans, “Selection of The Order of Autoregressive Models For Spectral Analysis of Doppler Ultrasound Signals”, *Ultrasound in medicine & biology*, Vol. 16, No. 1, pp. 81–91, 1990.

30. Haykin, S. and S. Kesler, *Prediction-Error Filtering And Maximum-Entropy Spectral Estimation*, Springer, 1983.