CAPTURE, ANALYZE AND DETECT MALICIOUS ACTIVITIES IN A UNIVERSITY NETWORK TRAFFIC

Thesis submitted in partial fulfillment of the requirements for the award of

Degree of

Master of Engineering
in
Information Security

Submitted By
Harleen Kaur Gill
801333006

Under the supervision of:
Dr. Maninder Singh
Associate Professor

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT
THAPAR UNIVERSITY
PATIALA – 147004

July 2015

CERTIFICATE

I hereby certify that the work which is being presented in the thesis entitled, *"CAPTURE, ANALYZE AND DETECT MALICIOUS ACTIVITIES IN A UNIVERSITY NETWORK TRAFFIC"* in partial fulfillment of the requirements for the award of degree of Master of Engineering in *Information Security* submitted in Computer Science and Engineering Department of Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of *Dr. Maninder Singh* and refers other researcher's work which are duly listed in the reference section.

The matter presented in the thesis has not been submitted for award of any other degree of this or any other University.

Harleen Kaur Gill

(Harleen Kaur Gill)

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

(Dr. Maninder Singh)
Associate Professor,
Computer Science and Engineering Department

Countersigned by:

(Dr. Deepak Garg)
Head
Computer Science and Engineering Department
Thapar University
Patiala

(Dr. S. S. Bhatia)
Dean (Academic Affairs)
Thapar University
Patiala

Acknowledgment

I would like to thank a number of people for their help and unending support over the year. Most of all, I would like to thank my supervisor, **Dr. Maninder Singh**, not only for his paramount mentorship but also for his guidance, patience and kindness to me. I owe him a heartfelt gratitude for galvanising my interest in the area of network security.

I am also thankful to **Dr. Deepak Garg**, Head of Department, CSED and **Ms. Jhilik Bhattacharya**, P.G. Coordinator, for the motivation and inspiration that triggered me for the thesis work.

My parents have always been the source of courage and inspiration for me. I am thankful for their unconditional support and love.

I am indebted to my brother **Lovleen** for being a motivator and for eye-opening conversations.

I highly acknowledge my friends for their kind words and everyone who has given a well meant prod in the posterior!

Harleen Kaur Gill

801333006

Abstract

In this thesis, we delve into the patterns of university network traffic and present the issues from an empirical aspect. Distinctively, this research capitalizes on hand-classified Internet traffic. It is crucial to understand patterns of university traffic and usage behaviour of end users. We address the problem of identifying malicious activities and understanding Internet usage within the university campus. This thesis aims at discovering the hidden patterns based on the analysis done on the captured traffic. To tackle the problem, systematically traffic is captured, filtered, managed and then analyzed. This approach gives analysis based on some python scripts and some open source tools which gives flexibility for distribution and code modification.

Signature based IDS require previous database of the anomaly patterns so that it can detect the attacks based on that information. On the flip side, attacks develop gradually to circumvent detection from signature based IDS. Another solution is to depend on statistical network traffic analysis. We have opted for the latter solution. In this manner it is possible to timely recognize abnormal network behaviour. Monitoring the network traffic is of prime importance for network security as it provides information regarding security breaches and helps to understand their impacts. Network monitoring is helpful in gathering useful information for security managers, network managers, marketing personnel, planners and others.

Table of Contents

List of Figures

List of Tables

Abbreviations

ACK	Acknowledgement
ARP	Address Resolution Protocol
ARPANET	Advanced Research Projects Agency Network
CLI	Command Line Interface
CV	Cross Validation
DCE	Distributed Computing Environment
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
DoS	Denial of Service
FCAPS	Fault Configuration Accounting Performance Security
FQDN	Fully Qualified Domain Name
FTP	File Transfer Protocol
GPL	General Public License
GUI	Graphical User Interface
HIPAA	Health Insurance Portability and Accountability Act
HTTP	HyperText Transfer Protocol
HTTPS	HyperText Transfer Protocol Secure
ICMP	Internet Control Message Protocol
IDL	Interactive Data Language
IDS	Intrusion Detection System
IETF	Internet Engineering Task Force
IGMP	Internet Group Management Protocol
IP	Internet Protocol
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
ISP	Internet Service Provider
LAN	Local Area Network
LLC	Logical Link Protocol
MAC	Media Access Control
NetBIOS	Network Basic Input/Output System
OSI	Open Systems Interconnection

RFC	Request For Comment
RPC	Remote Procedure Calls
SOX	Sarbanes-Oxley Act
SSH	Secure Shell
SSL	Secure Sockets Layer
STP	Spanning Tree Protocol
SYN	Synchronization
UDP	User Datagram Protocol
URL	Uniform Resource Locator
VoIP	Voice over IP
WWW	World Wide Web

CHAPTER 1
INTRODUCTION

If you spend more on coffee than on IT security, you will be hacked. What's more, you deserve to be hacked. — White House Cyber security Advisor, Richard Clarke

1.1 Background

1.1.1 Usage of Internet

In the contemporary world, usage of Internet has increased drastically. This scenario is due to the fact that Internet provides a plethora of services to the users. Stating from leisure activities like playing games to important transactions, Internet has become an essential part of the lives of masses. Increase in demand of applications like Skype, Facebook, etc has elevated the network traffic at an alarming rate [1]. In addition to that, traditional methods of shopping and trading are also being replaced by Internet [2]. Financial institutions are relying progressively on the Internet for trading of commodities and equities. Now-a-days, people prefer shopping and trading online because of ease of use and less time consumption. Ultimately this has lead to lots and lots of data exchange and thus enormous network traffic. In the near future, communities and online business is going to emerge and ultimately web will continue to enlarge in content and features [3]. Therefore, it is crystal clear that humanity will also keep on maintaining its dependence on the Web [10].

1.1.2 Internet traffic

Describing the Internet traffic is not an easy task. In fact it is an open area for research yet. Characteristics of traffic vary according to the way the traffic was investigated. Features of the Internet traffic change with the introduction of some new application or with the change in user behaviour. Nearly two decades back about 75% of traffic used to be web traffic [5]. Traffic volume has increased drastically from that time and although the proportion of web traffic is still a lot in total Internet traffic but now-a-days applications like file sharing dominates the overall traffic [44]. Moreover, video distribution over IP is wide spreading and is increasing its contribution in the Internet traffic.

1.1.3 Monitoring as a part of information assurance and network security

Information assurance is a pursuit of reassuring information and handling perils related to processing, use, transmission and storage of data or information [6]. It is a broad term which includes protection of data in the terms of integrity, confidentially, availability, non- repudiation and authenticity.

- **Integrity**

  Integrity is maintaining trustworthiness and accuracy of data. To maintain the integrity, data can only be altered by authorized people. This can be done with the help of file permissions and access controls. Data integrity is not limited to saving the data from getting altered by non-trusted parties, but it also includes saving the data from errors introduced by degradation [41].

- **Confidentiality**

  Confidentiality is somewhat same as privacy. Confidentiality is to make certain that information is accessible to authorised users only. It is a measure to prevent the wrong people getting access to the sensitive information. In network security confidentiality is mostly attained with the help of cryptography.

- **Availability**

  Data is only useful if it is available when required by the authorized and authenticated user. It is important to provide adequate bandwidth and prevent occurrence of bottlenecks.

- **Non-Repudiation**

  Non-repudiation means all the parties involved in the transaction should be aware of it. Sending party or receiving party cannot deny its action once the data is sent or received.

- **Authenticity**

  Authenticity is ensuring that documents, data, transactions or communications are genuine. It is validating that parties involved are the same who they claim to be. This can be done with the help of "digital signatures". Digital signatures ensure that data is sent by somebody having signing key and hence is genuine. It prevents the user or information from being forged.

An acceptable example of the features above would be an online transaction. In this example, integrity is achieved by TCP's checksum and re transmission

algorithms, confidentiality via RSA cryptography and authentication with SSL certificates. These all things are done through the use of HTTPS protocol. Availability is managed by the infrastructure of organization and non-repudiation is handled at application level.

1.1.4 Monitoring as a part of network management

FCAPS defines 5 areas of scrutiny for network management [22]:

- **Security management**
  It is mainly concerned with regulating access to the network and to the other devices.
- **Configuration management**
  It is primarily concerned with auditing network devices, tracing the alterations done to configuration. It also permits planning for future development.
- **Performance management**
  It includes determining metrics such as response time, utilization and error rate for existing infrastructure.
- **Fault management**
  In terms of network management, a fault has negative effect on the network. Fault management plays the role of predicting such events wherever possible.
- **Accounting management**
  It deals with managing the users in terms of providing them authentication, authorization and backup.

1.1.5 Security breaches

With the increase in usage of Internet threats to security are also increasing. There is always the risk of invaluable credentials getting stolen [10]. The extremity and measure of attacks organized against network infrastructure have escalated in recent years [13]. There are several threats to the safety of Internet. Some include security of hosts, some threats are concerned with privacy of data, others involving danger to the infrastructure of networks.

In the earlier times, orchestrating an attack was not so easy. Attacker needed lots of experience and intelligence. Usually the motivation was fame and curiosity [10]. Now-a-days, motivation behind the attack is generally financial gain. There are

various toolkits available to launch a lot many attacks. These Exploit Kits are developed by experienced attackers which are further used by novice people to launch attacks [11]. With the purpose of challenging detection mechanisms, Exploit Kits are periodically upgraded with new evasion techniques [12].

1.1.6 Network threats

As the thesis is based mainly on the security of the network so we will only discuss network threats in detail. Network Threats are the pre-eminent focus of detection in our research. This section will elaborate a plethora of prevailing network threats. Network threats are numerous and are of a variety of types. They range from single host involved in the attack to hundreds and thousands of hosts being indulged. In this thesis bounded summary of threats ubiquitous these days is presented.

1.1.6.1 Worm threats

Virus is a code that replicates itself and worm is an extension of virus. Worm has a mechanism to propagate between hosts. They can be characterized by their capability to expand directly or indirectly from one host to another [23]. Various stages of propagation of the worms are:

- **Target discovery**

  Target Discovery is done by active scanning or passively with pre-generated lists. Although many worms performs this step of discovering the targets but there are some worms that targets randomly generated hosts without the proper mechanism.

- **Transmission**

  Transmission of worm is attained either by using some prevailing communication mechanism, such as email or by some known susceptibility on the target host. There are some worms which uses secondary channel to send the worm code. Example of such a worm is blaster.

- **Execution**

  Execution or activation of the worm can be attained via direct or indirect human intervention. Direct human action can be the click of mouse and indirect human action can be machine reboot. Worm activation can also happen by self execution or via scheduled process activity.

1.1.6.2 Trust abuse threats

In this research this category is not addressed directly but it needs to be discussed for the sake of knowing the threats. Trust abuse threats do not have much effect on network performance [20]. In essence, trust based attacks are informational attacks. The motive of these attacks is to gain some valuable information from the user. They are most common through emails and referred to as phishing commonly. Mostly the attacker tries to get access to sensitive information like details of credit cards and passwords for various accounts like eBay [25].

1.1.6.3 Denial of service

In this type of attacks, the attacker misuses the Internet connectivity to paralyze the services provided by the victim site [26]. DoS attacks can prove very fatal to the victims. First attack of significance was orchestrated by Robert Morris. It was launched on Nov. 2, 1998. It was said to crash some 5000 machines for several hours [27]. DoS attacks can vary in many ways. They can use spoofed or genuine addresses; they can use different data rates; number of hosts can vary as well as the control topology and they can last from seconds to long period of time [28].

The attacks can be categorized based on the way by which DoS is achieved:

- **Flood based attacks**

  They lean on exploiting the resources by transmitting a huge amount of traffic. This type of attack prevents the genuine traffic from reaching the target host as all the bandwidth is consumed by sending enough UDP packets. It is not necessary that the packets should be UDP, it could be any protocol [28].

  In this case, mitigation is comparatively simple. If all the traffic from the attacker destined for victim is blocked, then normal services can be revived. Mitigation in this attack could be hard by the use of IP spoofing. Spoofing is a casual term meaning forging. In IP address spoofing, host replaces source IP field of its packet with another host's IP address. By doing so it appears that the packet has been sent by some another host.

- **Multiplier based attacks**

  They are the enhancement of flooding attacks. Here the attacker uses various methods to elevate the load on target. This proliferation in load is done via secondary mechanism. Replacing the source IP with the target in broadcast

ICMP echo request would be a perfect example of this. All the active hosts will receive the echo request and then will reply to the target. This will result in generating a large multiplier. Ultimately the bandwidth of the victim will be flooded with the modest amount of traffic initially sent by an attacker.

- **Service vulnerability attacks**

  These attacks make use of the potential multiplier within a service. An example of this is connection limitation within TCP based services. Suppose a TCP server permits 15 connections with each connection having timeout of 3 seconds. Then connection limit can be reached by not responding to at least 5 'synchronize' packets per second at the target [28]. Another classic example of this can be calling a processor intensive server side operation within web server repeatedly. This denies processor time to the genuine users effectively.

1.1.6.4 Probing

Various vulnerabilities can be found in the network by probing. These vulnerabilities can be further used by the attacker to attack the network. There are various utilities which can help the attacker to scan the network by checking firewall logs, auditing policies and also by checking patches. There is mainly two type of probing:

- **Enumerating**

  In this the attacker tries to find out shared folders, user accounts, etc on the target network. This information can be helpful in exploiting the network and even the sensitive information of the users can be leaked.

- **Port Scanning**

  By port scanning open and vulnerable ports can be known. Open ports can tell a lot about the target system and the services it is using.

1.1.7 Three D's of network security

Network security can be defined by elaborating it into three D's:

- **Defence**

  It is most important form of network security. When the attacks are defended, then there are lesser chances of network performance getting degraded [39]. Along with that it also reduces the chances of other assets getting

compromised. Network defences incorporates methods such as spam or virus filters, firewalls and router access lists.

- **Deterrence**

  It is the second mode of security. Deterrence in network security can be referred as the measures taken to resist the attacks beforehand. It demands laws for the users so that organization's security policies may not be violated. Deterrence minimizes the risks to the information through fear.

- **Detection**

  With the help of detection, security compromises reduce and damage can occur only for limited duration if proper security measures are taken in the timely manner. It includes IDS, log files, etc.

1.1.8 Traffic analysis

Traffic capture gives information for configuration, performance, security and fault management. However, in our thesis we are mainly concerned with security management. DoS attacks indirectly affect the performance of network.

Researchers are always keen to know the underlying traffic demands to hike the profit by completely utilizing the network resources [1]. Although network traffic analysis is becoming more challenging with each day but it is of prime importance in computer security. Analysis of traffic is becoming troublesome because hosts are elevating in number and diversity, network traffic is proliferating in complexity and volume, and attacks are mushrooming in sophistication and variety. Therefore it is vital to identify normal and malicious traffic patterns as the results can be used to monitor unseen traffic.

In this field, research has laid emphasis on utilizing machine learning algorithms to differentiate between normal and malicious traffic. Machine learning algorithms are helpful in this area as they automatically educe the peculiar features of normal traffic and the anomalies from the huge data. Hence, they yield highly precise results with little manual intervention [7]. Another advantage of machine learning algorithms is that they have a high speed of detecting anomalies. Machine learning algorithms identify drastic changes in patterns of traffic very quickly [8] [9].

University networks have no strict regulations compared with enterprise networks, so they are more prone to attacks and hence needs to be monitored. For e.g.

businesses employ administrators who will make certain that safety on business hosts is in accordance to the decided policies [3]. Such a scenario is not possible in campus networks. Moreover, in enterprise networks various restrictions can be imposed for the usage of Internet on employees whereas in universities students use their own laptops which could be already infected and hence can cause threat to the whole network. The framework that Internet utilizes has been apparently vulnerable to attack, especially for networks in distributed environments as that of universities. Henceforth, security within university networks is paramount.

Usually flow information is utilized for traffic monitoring [4]. In this thesis, database was built from live data. Then with manual inspection classification of data is done alongside laboratory investigation. A vast understanding of the field along with general literature is also required.

However, it is not an easy job to measure and analyze the university traffic. There could be many hindrances as mentioned below:

1.1.8.1 Challenges while measuring university network traffic

There were certain problems faced while capturing the university traffic. Some of them are explained below:

- **Privacy issues**

  Users are concerned with the fact that their personal details should be secure and should not be available to the third parties. Although traffic collected for analysis should be meaningful but it should not contain sensitive information.

- **Unwanted data**

  It is not always necessary that captured data is always what it supposed to be. Many a times we are not able to measure traffic of interest directly.

- **Enormous traffic**

  Traffic collected is of huge volume and it is challenging to deal which extensive data. Much effort is required to store traffic and manage traffic with high volume.

1.1.8.2 Challenges while analysis of university network traffic

Certain hindrances while analyzing the university traffic are described below:

- **Data encryption**

  Data is being encrypted these days before transmitting to secure it from getting stolen. As the usage of Internet is elevated in every field, more and more confidential data is there on Internet which needs security [40]. People want their documents to be secure on Internet. Even security is a big concern in e-commerce sector as well. Secure online transactions are required. Therefore, for security purposes data is encrypted and it is a hindrance in analyzing traffic.

- **User behaviour**

  Sometimes which monitoring the traffic we have to consider user behaviour as well. There may be some new application launch at a particular time like Facebook or may be YouTube which may lead to change in behaviour of user [57]. Such scenarios should be considered while analyzing the traffic and should not be confused with malicious activities on Internet [32].

1.2 Motivation

Importance of university network traffic analysis may affect many areas. Various trends prevailing among students can be known by analyzing the traffic. This would help in enhancing the various and facilities and the bandwidth requirements within the campus. It is possible to know that more traffic is going to educational site or to the other sites. Results may also be helpful in determining various attacks on the network and hence various techniques can be used to tackle with them. Even the sources of attacks can be detected with the analysis and accused may be punished accordingly by the university authorities.

1.3 Overview of thesis

This work contributes to analysis of university network traffic for the security purposes.

Chapter 1 describes the basic introduction to security and describes network security in detail. It also elaborates the challenges while measuring and analyzing the traffic.

Chapter 2 presents the related research in the area. It describes various techniques used till date for traffic analysis to know the trends within a network and also discusses various methods of detection of malicious traffic. This chapter also elaborates the techniques to detect DoS and probing attacks.

Chapter 3 gives the problem statement and elaborates the need for this research. It describes the main objectives of the thesis.

Chapter 4 describes the way to capture the data which was used in the research. It also explains the methodology and various machine learning algorithms used for analyzing the traffic.

Chapter 5 elaborates the patterns, relationships and malicious activities found in the traffic. It tells the better approach for detecting malicious activities and represents some other figures.

Chapter 6 infer conclusions from the research work and finally discusses the likelihood of further investigation.

Chapter 2

LITERATURE SURVEY

In this chapter, other related research in traffic analysis field is examined and inferences are discussed. Network traffic monitoring has become a wide area of research due to the elevation in number of security threats. Here, we have represented the research most relevant to our thesis. The focus of this chapter is mainly on feature selection, traffic capturing, monitoring and detecting denial of service as well as probing attacks.

2.1 User behaviour

Maria *et al.* [37] presented the user behaviour characterization from an Internet Service Provider standpoint. The user behaviour was characterized based on the criteria which included:

- User request patterns
- Session arrival process
- Bytes transferred within a session
- Session duration

They also monitored the e-business activities of the users. Users were categorized into residential users and small offices and analysis was done on both the categories separately. They found that session arrival times were comparatively high during the day time. Their classification only relied upon the traffic collected at specific interval of time.

Considering the requirement for traffic monitoring and analysis Kim *et al.* [18] examined the features of Internet traffic from the viewpoint of flows. They concluded that the occurrence of flash flows affects the working of prevailing flow-based traffic monitoring systems. Some of their main findings while analyzing the IP traffic were:

- Packet count as well as byte count of TCP traffic was larger than UDP traffic. On the other hand, flow count of UDP traffic was double to that of TCP traffic.

- Most of the flash flows were UDP flows. The percentage of UDP flash flows was 76.8% which is very large compared to that of TCP flash flows which was only 7.7%.
- Peer-to-peer applications were mainly responsible for most of the flash flows.

Overall they concluded that flash flows with small size and short duration were important to be considered for traffic analysis system performance. Their work did not discuss anything about the nature of flash flows of malicious traffic.

Färber *et al.* [31] discussed the behaviour of home users in dialup sessions. They presented the typical characteristics of traffic based on the log data of University of Stuttgurt. Their results represented variability of holding time, long holding times and inter arrival times of the users. They proved that the user behaviour is very much influenced by telephone tariff scheme.

2.2 Traffic classification

Many studies reveal that a small proportion of traffic type consumes most of the bandwidth. Therefore, it becomes vital to know the characteristics of such traffic flows for proper traffic management and to fulfil further bandwidth requirements. Heidemann *et al.* [13] studied the traffic features like burstiness, size, rate and duration and also found the correlation between them for traffic engineering and modelling purposes. The main contributions of their paper were:

- Characterizing prior definitions in a systematic manner for the properties of heavy-hitter traffic.
- Proving correlation between combinations of rate, burstiness and size.
- Explaining correlations by application and transport-level mechanisms.

The results of their research were based on very limited traffic traces which was one of the main limitations of this paper.

Zuev *et al.* [14] used supervised Bayes estimator for traffic classification. The network data which acted as input to this procedure was hand classified. They were able to acquire more than 83% of accuracy for both per-packet and per-byte criteria. They maintained the access to the full payload as it could be the only sure method to characterize network applications according to them. They said that it was quite tough

to maintain the access to full-payload trace due to legal and technical constraints, so many a times header-only trace are available. Therefore the data collected without any restriction, can act as training input for statistical classifier which can further provide approximate accurate results for classification of the traffic whose only headers are available. This technique could help to classify the traffic with less available information but on the flip side it was quite complex.

Traffic classification is mandatory to know the statistical information and average of the traffic crossing a hub or a pipe. Sasan Adibi [53] did an analysis to know various insights of traffic like average packet size, traffic on a particular link and average end-to-end delays. Such information could be of great help to design robust networks and avoid congestions. His paper included application, packet and flow based aspects to classification of traffic.

Concept of machine learning was started being used to classify the traffic. McGregor *et al.* [51] proposed a method to break the traffic into clusters with each cluster having different characteristics. This clustering was done on the basis of machine learning techniques. Mainly the clusters include single and multiple transactions, bulk transfer and interactive traffic. The clustering of traffic was done with the help of tool called as Weka.

Moore *et al.* [34] illustrated one of the applications of *supervised* learning. They categorized internet traffic by application on the basis of Naïve Bayes estimator. With the help of Naïve Bayes algorithm they were able to illustrate high level accuracy. They not only analyzed the performance on the basis of accuracy but also took factor of trust into account.

Erman *et al.* [33] researched on classification of network traffic based on clustering algorithms. They thought that classification on the basis of payload and port has become extremely tough because of encryption and masquerading techniques. As a result, they classified the traffic by utilizing distinctive features of applications while communicating on the network. They only used transport layer statistics to cluster the same traffic in same group. Two unsupervised algorithms were utilized by them i.e. DBSCAN and K-Means. They used empirical Internet traces to compare their results with already used AutoClass algorithm. Finally they proved that their results with K-Means and DBSCAN were quicker than AutoClass. Their results indicated that although DBSCAN has comparative lower accuracy but still it generates better clusters.

Job of network management has become more challenging with the increase in flexibility in networks. To lay emphasis on flexible networks, now any host can transmit any kind of information at any time. Considering this Estan *et al.* [56] introduced a methodology for characterization of traffic that automatically grouped it into clusters of conspicuous consumption. Their approach dynamically produces hybrid traffic definitions instead of static analysis. Their technique can also prove useful in distinguishing new traffic patterns, such as peer-to-peer applications and network worms. They also proposed a prototype system called AutoFocus and used it to find the unusual and dominant modes of usage on a numerous production networks. This paper introduced method for multidimensional traffic clustering and the usage patterns which are driving traffic growth. AutoFocus automatically extricates patterns of resource consumption based on traffic log of single link and in a single interval of time. This tool could be extended to extract patterns across time and space.

Another method of classifying the Internet traffic was proposed by Karagiannis *et al.* [35] to categorize the traffic based on the applications that generate them. This method observed the patterns of host behaviour. It identified the traffic flow at the transport layer. The important features of this approach were that it did not have any access to packet payload or the information about port numbers. Due to these restrictions privacy was respected. Moreover their approach could be adjusted to make the balance between accuracy of classification and traffic flows that were successfully classified. Their results were able to categorize about 85% of traffic with 95% accuracy. The approach introduced by them was called as BLINC. They argued that noticing the actions of host gives more information and could unveil the behaviour of applications of host. On the other hand this approach had some limitations as well. BLINC was not able to figure out sub-types of specific applications. Moreover, this technique was able to classify encrypted traffic only when it was limited to transport layer.

2.3 Traffic measurement and analysis

To get the information of characteristics of traffic, its measurement and analysis is essential. Therefore, this area of research has drawn the attention of researchers from many years. Depending upon their requirements researchers has analyzed the traffic

accordingly. Some have analyzed the traffic for short period and others made a long term investigation.

Cho *et al.* [29] presented the way to establish repository of traffic data which contained information of backbone traffic. Traffic was collected by *tcpdump*. The traffic traces which were captured were made public after removing sensitive information. This was done to make the data sets freely redistributable so that research on traffic analysis can be promoted. The repository also proved useful in development of various traffic analysis tools.

Borgnat *et al.* [28] collected network traffic for a complete period of seven years and one day to predict the evolution of traffic. They investigated the characteristics of traffic both for application usages and at TCP/IP layers. After analyzing seven years long dataset, researchers concluded that traffic statistics unveil a huge variability, mainly because of traffic condition variations like restrictions and congestions. The statistics also reveals the occurrence off anomalies in a random manner. In the recent years the idea of self similarity in used in analyzing the traffic. Crovella *et al.* [30] has shown the evidences that the subsets of traffic due to World Wide Web transfers shows features that are unvarying with self similarity. They closely monitored the dependence pattern of WWW traffic.

While the proper service of Internet has reached the prominent parts of the world, but the rural areas of many under developed countries are still disconnected. Although efforts are being made to provide connectivity but many networks even fail to provide quality service needed for even simple applications. Johnson *et al.* [31] investigated the problem in depth and worked on the network traces from rural areas of Macha and Zambia. Their research revealed that traffic in rural areas has a major difference from that of the developed world. They proposed to take into consideration both social and technical factors while designing a network. They even investigated for malwares in the traffic as it is challenging in rural areas to download virus signatures because of scarce satellite bandwidth. Analysis of malwares in such areas is also necessary because of lack of awareness and people use more vulnerable operating systems. For detection of malware in traffic snort was utilized by them. Limitation of this paper was that it was not able to detect virus payloads as it requires deep packet inspection.

Then further Kumar *et al.* [36] worked on this implementation. They implemented the technique of Deep Packet Capture and Deep Packet Inspection to

effectively analyze the packets. Their approach was useful to monitor the activities in private and public network. With the help of Deep Packet Capture network packet payload, traffic crossing the network was captured. DPI was performed to verify the packet data and to reveal the cause of network problems with the help of forensics analysis. It also helped in uncovering security threats.

It is vital to know the usage of Internet to fulfil the future demands of users. Considering this Kihl *et al.* [37] analyzed traffic measurements and derived user behaviour models from Swedish municipal broadband network. They illustrated the Internet usage in terms of volumes, patterns and applications. In their paper, other user activity characteristics like session length were also analyzed. They used a tool known as Packet Logic to perform the measurements. Finally it was depicted that the results for user session length were different from the ones that were traditionally assumed.

Smith *et al.* [54] proposed a method for Internet traffic analysis which was based on statistical cluster analysis and application-level communication. They claimed that the method can be a foundation for constructing flexible traffic generation tools. They believed that traffic workloads for simulation and testing should reflect the distinguished patterns of communication.

Henrik Abrahamsson [38] researched on Internet traffic management and took three aspects into consideration i.e. bandwidth allocation to TCP flows, web traffic modelling and traffic engineering. His work represented a blueprint of web client traffic. He derived probability distribution which described data transferred with one user request and session lengths. He used ns-2 simulator for experimentation purposes and figured out the number of users that can be handled by a particular network.

While a lot has been done to analyze the offline traffic but comparatively less research is there in analyzing the real time traffic. This is upcoming area of research and is quite challenging. Robert T. Braden [55] developed a flexible packet monitoring algorithm which operates in real time. His program analyzed the traffic patterns and collected statistics on a packet network. He designed the packet monitoring program called *statspy*, which was a settlement between generality and efficiency. Generality of *statspy* was important because the people who used the algorithm asked complex questions about traffic patterns.

2.4 Detecting malicious activities in traffic

As the cyber-attacks are increasing these days, so it has become important to develop techniques that can distinguish significant communication patterns to detect malicious activities. Bhattacharya *et al.* [16] developed a technique to build comprehensive behaviour profiles in terms of communication patterns of services and end-hosts. Their methodology relayed entropy-based and data mining techniques. The significant features of their methodology were:

- Automatic behaviour classification
- Significant cluster extraction
- In-depth interpretive analysis by structural modelling

Their approach utilized the flow-level header for analysis and therefore can be extended to analyze application-level payload transported in IP packets.

Diaz *et al.* [39] analyzed pcap files with the help of some basic Linux tools. They examined the traffic with the help of tools like tcpdump and other tools which can read pcap data. They found the abnormal traffic in the Internet by utilizing the RFC, which is a memorandum published by IETF describing innovations, behaviours or methods applicable to working of Internet-connected systems as well as Internet. RFC is produced for every network protocol that tells how a particular network application should work. These descriptions are honoured by most of the commercial software. With the help of RFC it becomes easy to distinguish malicious traffic from that of the normal traffic as normal traffic complies with RFC and abnormal traffic does not [47].

Meghanathan *et al.* [48] discussed the techniques and tools available to perform network forensics. They discussed the tools like Web Histogram, eMailTrackerPro, Ethereal, etc. Furthermore, they did a survey of various IP traceback techniques which helped in identifying the origins of attacking IP packets. Lastly, use of Honeypots and Honeynet Architecture was also discussed by them. Their paper did not explain any technique regarding wireless network forensics.

Wang *et al.* [49] represented a method for network forensics analysis which had evidence presentation and automated reasoning. They proposed evidence graph to facilitate the presentation of intrusion evidence. They also developed a hierarchical framework for automated evidence analysis. It included global and local reasoning.

Global reasoning identified the group of correlated hosts in attacks and local reasoning inferred the role of the suspicious hosts. They merged analyst feedback to automated reasoning process with the help of evidence graph model.

Darren Manners [58] tried to find out the malicious activities happening within the organization. In his research work, he utilized the user agent field to detect abnormal traffic. A plethora of free tools like tshark, tcpdump and wireshark were used by him to separate the normal traffic from malicious traffic. He examined the user agent fields in the HTTP request header to detect attacks. He proved that there is real threat to the organizations that are not alert while dealing with user agent data.

Traceback is the only solution to hold the attackers liable for their deeds. Mitropoulos *et al.* [46] presented features of various traceback mechanisms and then proposed a method to provide information for Digital Forensics by the amalgam of the existing traceback methods. Their method did not analyze the traffic between cross-administrative domains.

Matthew V. Mahoney [17] claims that benign traffic can be distinguished from hostile network traffic as its nature is different. He proposed an anomaly detection system which had two stages:

- In the first stage traffic was filtered to pass the packets of most interest.
- Second stage modelled the common protocols like IP, TCP, FTP, SMTP, HTTP, etc at packet byte level to flag events.

He claimed to detect 132 attacks from total 185 attacks in the network but his model failed to detect the nature of attacks. He was just able to find the interesting and unusual patterns from large amount of traffic.

According to Barford *el al.* [19] recognizing the anomalous behaviour of traffic was usually dependent upon ad hoc methods which were developed by years of experience. In their research work, they presented characterization of abnormal traffic behaviour. To analyze the traffic traces they used the open source tool called FlowScan. They identified the statistical properties of anomalies in their work.

Vern Paxson [21] described Bro, which was a tool to detect attackers in the network in real-time. Bro works by monitoring the network link in a passive mode. They designed a system capable of monitoring the traffic at high-speed and gives real-time notification. To fulfil these results Bro was divided into two parts:

- **Event Engine:** Compressed kernel-filtered traffic to series of higher-level events.
- **Policy Script Interpreter:** Interpreted event handlers which were written in specialized language.

Their system is available freely in the source code form. It is only able to analyze four applications so far:

- FTP
- Finger
- Telnet
- Portmapper

Mostly the classification of network applications which were responsible for network flows relayed on some packet header fields or on the decoding of application layer protocol. These methods had some limitations. For e.g. if classification was done on the basis of port numbers then many applications may use unpredictable port numbers. In case if classification was done by protocol decoding then lots of computing resources is required. This becomes impossible if the protocols were encrypted or unknown. Considering this Zander *et al.* [52] proposed a methodology for application identification and traffic classification by using unsupervised machine learning. In their model flows were automatically classified on the basis of statistical flow characteristics. They found out the efficiency of their method by utilizing data from various traffic traces. They also used feature selection to search an optimal feature set. Their paper can be extended to figure out various security incidents like port scans.

It is critical to find anomalies in the network for end users and as well as for the network operators. To diagnose anomalies in the network, one needs to extract and then interpret the anomalous patterns from the high-dimensional traffic. Crovella *et al.* [15] proposed a method to diagnose anomalies from traffic by separating space occupied by network traffic measurements into subspaces corresponding to anomalous and normal network conditions. Their method was able to:

- Identify the origin-destination
- Estimate the traffic involved in anomalous flow

- Detect when anomaly is occurring

With the advancement in research, subspace method for anomaly detection was extended to the next level. Lakhina *et al.* [9] presented the large-scale exploration of subspace method when executed on traffic flow. This method merged information from flow measurements retrieved throughout the network. Subspace method was applied to three types of flow traffic:

- Packet counts
- IP-flow counts
- Multivariate series of byte counts

They proved that by applying subspace method on each type of traffic, brought into light different set of anomalies.

To keep the network secure, intrusion detection is very important. Xiang *et al.* [45] used the concept of data mining to ease the labour-intensive task of finding patterns in the traffic. They proposed a multi-level hybrid classifier. It used amalgam of clustering algorithms and tree classifiers to detect malicious activities within the traffic. Finally they compared the performance of their algorithm with other approaches like 3-level tree classifier, MADAM ID, etc and found the improvements with respect to false alarm rate as well as intrusion detection rate. They combined Bayesian clustering with decision trees to find to DOS and probing attacks.

Kumar *et al.* [50] analyzed NSL-KDD dataset to detect new intrusions by applying clustering techniques on it. They used K-Means clustering algorithm to categorize the probe and DoS attacks. Clustering techniques are useful in case of availability of a large volume of unlabelled dataset.

DoS is one of the major threats to the Internet today as its impact is severe. It exhausts all the communication and computing resources with no prior warning. Douligeris *et al.* [42] developed a structural approach to detect and defend the attack. They presented a clear view of the attack and proposed defence mechanism based on source network, victim network and the intermediate network. Lau *et al.* [43] were motivated from attacks on sites like Amazon.com, Yahoo.com, etc and worked on distributed denial of service attacks. They used ns-2 simulator to analyze that performance of queuing algorithms which were implemented in the network router.

They also examined that during the attack whether genuine user can get the required bandwidth or not. They found that queuing algorithms based on classes can provide bandwidth to certain input flows in case of persistent denial of service attacks.

Chapter 3

Problem Statement

Malicious activities occurring in the network can cause severe damage to the underlying network infrastructure. Measurement of the traffic from and to the base is required to understand the network behaviour. Monitoring university traffic is essential to save the users from unwanted problems like slow Internet speed and save the resources from getting saturated unnecessarily. Moreover, data like research work needs to be kept confidential and integrity of data also needs to be preserved. Otherwise, anybody could enter into the system and can change data like grades of the students. Although Authentication-Authorization-Accounting servers are used to verify the users before letting them in the intranet but students cannot be compelled to work on guest accounts, as done in enterprise networks.

Lack of anti-viruses in security posture may lead to infect individual's system which can further affect the network as well. Therefore, the machines within the campus and university networks are more prone to the attacks. Understanding the usage patterns is vital to know the future demands and shortcomings in the prevailing framework. Consequently, university network traffic needs to be observed. Although there are some references for securing the university networks but to our knowledge, not much has been done till date.

3.1 Objectives

- To capture and analyze network traffic in an academic setup.
- To extract protocol hierarchy and correlate its existence with benign and malicious traffic.
- To perform network traffic categorization.
- To validate proposed method of hierarchal classification with university network traffic dataset.

Chapter 4

Implementation

In order to detect malicious activities as well as to relate traffic patterns, we need to apprehend aspects of malicious traffic and user behaviour. In this chapter overall framework is described.

4.1 Analyzing and examining university traffic

When some activity of interest is identified, analysts extract network traffic to analyze it further and to determine the cause of occurrence as well as to know the way it affected the organization's network. This procedure could be very simple like reviewing some log entries and investigating if the event was true, or a complex sequence of examining dozens of sources and then manually determining the relation between sources and the significance behind the event. Knowledge of the campus environment is must. It is necessary to have information like network architecture. If the analyzer has the solid cognizance of typical patterns of network usage then it becomes easy to understand if something suspicious is happening in the network.

4.2 Weka

Weka is software written in Java which acts as a platform for applying various machine learning algorithms on a dataset. It was developed at University of Waikato in New Zealand. This software is free and is available under GNU. It helps in predictive modelling and data analysis with the help of algorithms and visualization tools contained in it [39]. Advantages of Weka include:

- Ease of use because it has Graphical User Interfaces.
- Under GPL Weka is available free.
- It is portable due to its programming in Java and hence can run on any computing platform.
- Comprehensive collection of data analysis and modelling techniques.

Various data mining tasks supported by Weka are data pre-processing, classification, clustering, regression, feature selection and visualization. Weka also

provides access to SQL databases with the help of Java Database Connectivity. Weka has four user interfaces:

- Explorer
- Experimenter
- KnowledgeFlow
- Simple CLI

Although main user interface of Weka is Explorer but same task can be done with component based KnowledgeFlow and from Simple CLI. Experimenter option allows you to compare predictive performance of machine learning algorithms on collection of datasets. The Explorer has several panels like:

- **Preprocess:** This panel has the functionality of importing the data from several sources. Then pre-processing can be done by using filtering algorithms. The filters help to transform the data in the required form. Various attributes and instances can be deleted according to specific criteria.
- **Classify:** This panel helps the user to apply regression and classification algorithms to the dataset. It also enables the user to predict the accuracy of predictive model and to conceptualize erroneous predictions.
- **Cluster:** It facilitates user to apply clustering techniques to the resulting dataset. The dataset gets distributed into clusters of same kind after applying clustering algorithms on it.
- **Associate:** This feature helps to find vital interrelationships among attributes in data by giving access to association rule learners.
- **Select attributes:** This panel figure out the most predictive attributes in dataset by providing various algorithms.
- **Visualize:** A scatter plot matrix is visualized with this panel. In the matrix, each scatter plot can be enlarged and can be scrutinized further with the help of selection operators.

4.3 Wireshark

It is a free open-source packet analyzer and cross-platform tool. Initially it was named as Ethereal and renamed as Wireshark in 2006. It is used in various fields like communications protocol development, network troubleshooting, traffic analysis and

education. With the graphical front-end, Wireshark has integrated filtering and sorting options [41]. Port mirroring helps to capture the network traffic to any extend.

Wireshark is a tool that understands different networking protocols and their underlying structure. It parses and represents the fields with their meanings as mentioned by different networking protocols. Wireshark can only capture packets on the networks that are supported by pcap. Some features of Wireshark are discussed below:

- It can read the data from already captured file of packets or can capture the traffic from live network connection.
- Captured traffic can be browsed through GUI or via terminal version called Tshark.
- Captured traffic can be programmatically edited and can be refined with the help of display filters.
- Plug-ins can be developed for dissecting protocols.
- VoIP calls can be detected from the captured traffic. Media flow can be played, provided the compatible encoding is used to encode it.
- It can be ensured that only trigger traffic appear with several timers, settings and filters.

4.4 Tshark

Wireshark has a non-GUI version called Tshark. Tshark comes along with Wireshark and is terminal oriented. It is designed for capturing and processing the traffic traces when interactive version is not available for necessary. This tool helps in performing network protocol analysis and can be helpful in dissecting already captured network traffic by utilizing a range of options available.

4.5 Python

Python is a general purpose high-level programming language. Its design philosophy gives importance to code readability. Compared to other languages like Java and C++, Python allows the programmers to write the code in fewer lines. It enables clear programs by providing constructs intended. Python has several features like automatic memory management and dynamic type system and has comprehensive and large standard library. Python is a multi-paradigm programming language due to its full

support to structured programming and object-oriented programming. A vital characteristic of this language is dynamic name resolution which is also called late binding. Due to this feature, variable names and methods are bounded at the time of program execution [42].

Python is highly extensible and can be embedded in applications that require programmable interface. It is designed with the intension of having clear visual layout. In python, usually English keywords are used instead of punctuations. It also has lesser number of special cases and syntactic exceptions than the other languages. Rather than using curly braces, python uses whitespace indentation to delimit the blocks. This feature is known as off-side rule. There are three types of development environments for python:

- **Command Line:** Python acts as a shell in this developing environment. Most of the python implementations can work as command line interpreter.
- **Integrated Development Environment (IDE):** Other shells like IDLE and IPython add features beyond that in basic interpreter. These features include retention of session state, auto-completion and syntax highlighting.
- **Browser Based IDEs:** IDEs does not only limit to the desktop version but there are browser based IDEs as well. Some of the examples include Sage and PythonAnywhere.

4.6 Python weka wrapper

Python weka wrapper is a package that allows running algorithms and filters in weka from Python code. Processes of weka are executed in Java Virtual Machine (JVM). Further, for starting, communicating and for shutting down of JVM javabridge library is used. Python weka wrapper facilitates a wrapper around the non-GUI functionality of Weka. Following are the requirements for working of python weka wrapper:

- javabridge
- pygraphviz
- matplotlib
- Oracle JDK

Code Snippet:

```
data = loader.load_file(data_dir + "traffic.arff")
data.class_is_last()

from weka.filters import Filter
remove = Filter(classname="weka.filters.unsupervised.attribute.Remove", options=["-R", "1
3"])

cls = Classifier(classname="weka.classifiers.bayes.NaiveBayes")

from weka.classifiers import FilteredClassifier
fc = FilteredClassifier()
fc.filter = remove
fc.classifier = cls

from weka.classifiers import Evaluation
from weka.core.classes import Random
evl = Evaluation(data)
evl.crossvalidate_model(cls, data, 10, Random(1))
```

4.7 Experimental setup

In this thesis, traffic used for analysis was collected with the help of tcpdump. We have used Python script to serve this purpose. Python is a programming language which is helpful in many application fields. Moreover, Python and its APIs are open source unlike MATLAB. Various python API's like matplotlib, Mysql, numpy, pandas, etc. have also proved very useful. Another dataset was collected with the help of libpcap library. Figure 4.1 shows the environment where traffic was captured. University has two Internet Service Providers which are connected with Unified Threat Management system which is further liked to a layer-2 device. Port mirroring is done on the layer-2 device to send copy of network packets from the ports to network monitoring connection. Finally a capturing device is used to capture the traffic.

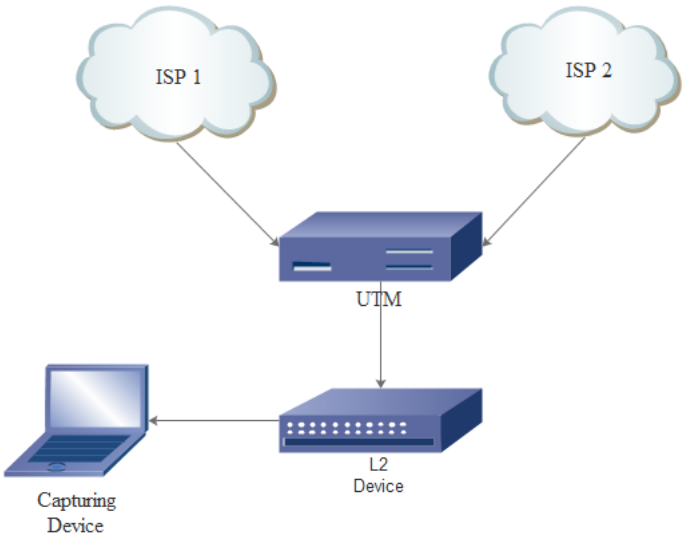

Figure 4.1: University network traffic capture environment

University network traffic was then filtered according to the requirement. This filtration of traffic was done with the help of command-line tool called Tshark. Different types of analysis require a different dataset containing parameters of relevance. Datasets in the required format were stored separately in the form of tables. As shown in Figure 4.2 data was captured and then traffic flows were stored.

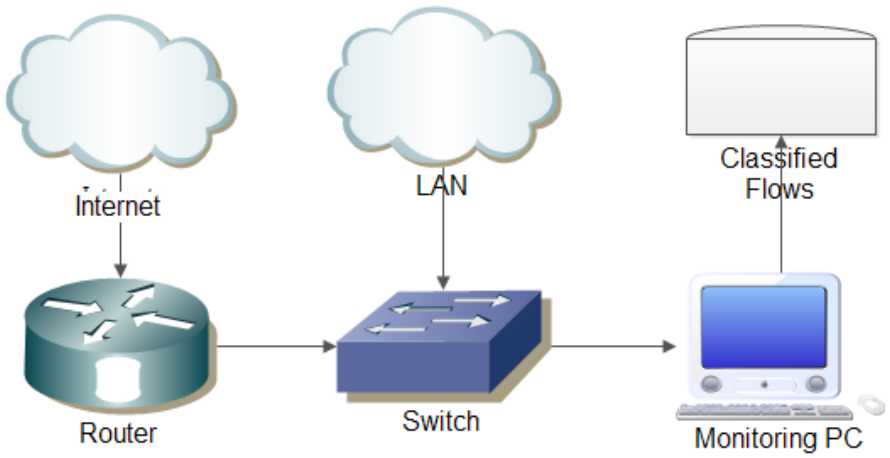

Figure 4.2: Storage of traffic flows

Further analysis was done by writing the scripts in Python. There were several tools which were involved in our research including Weka [24]. Weka is Waikato Environment for Knowledge Analysis. It is software available under General Public License. Tshark was used to explore the TCP packets in detail.

4.8 Methodology for predicting user behaviour

Firstly in this thesis, usage behaviour was analyzed in the University traffic. Current trends among the users were predicted based on the analysis. Usage behaviour was determined by analyzing the traffic in three ways:

- Protocol Hierarchy Statistics
- Web Categorization
- Hourly network usage patterns

4.8.1 Protocol hierarchy statistics

The communication between the machines is done by a set of predefined rules called protocols. Most of the networks are ordered systematically in levels or layers. Each layer provides certain set of services to the higher layers in the hierarchy. We filtered the traffic to get the protocols used in the communication. We analyzed the usage of protocols in hierarchical manner to see the percentage of use for various protocols.

4.8.2 Web categorization

After collecting the traffic, we applied filtration to separate the FQDN's from the University traffic. There are various databases available on the Internet to categorize these FQDN's. Then we compared the FQDN's to already defined databases to categorize them into eight categories namely:

- Shopping
- Entertainment
- Information Technology
- Business and Economy
- Advertisement
- Social Networking
- Sports
- News

4.9 Methodology for analyzing malicious activities

Incontrovertibly, to present the problem of deciding whether traffic is malicious or not indicates the existence of concept of normality. Normality in this research was

considered as the regular patterns of traffic. An event was enumerated as malicious because of its prominent deviation from its characteristic behaviour.

Traffic monitoring is needed to be done to keep a check on the students as well as on the employees, so that rules and regulations of the campus are properly followed. Henceforth, analysis was done on microscopic level i.e. analyzing packets and as well as on macroscopic level i.e. finding patterns in the traffic flow to detect the various attacks as shown in Figure 4.3. Instead of port definitions, deep flow as well as deep packet inspection was done.

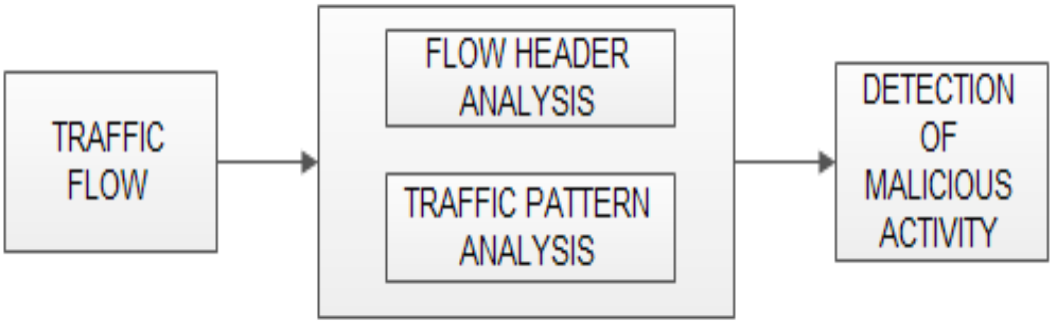

Figure 4.3: Detection process

In reference to network management, recognizing malicious traffic is frequently based on the experience gained from years. In our work, we have analyzed the traffic patterns as well as packet headers to recognize malicious traffic.

4.9.1 Detection process

- **Traffic capturing:** University network traffic was captured to analyze the prevailing trends among the users and to detect the malicious activities. Data set includes the university traffic collected at the particular intervals over a time period of seven days. The traffic capturing was limited to 20,000 packets per day. This was done considering the space constraints. Wireshark was used to capture the traffic and the output was stored in the form of pcap files.
- **Attribute selection:** It was an important part of the overall process because more number of attributes made the problem more complex to solve. Many of the traffic features did not contribute to the correctness of the classification task but in fact hampered it. Moreover, if the number of attributes rose then training time for machine learning algorithm was also proliferated. Therefore,

proper attribute selection was required. Selection of parameters significantly reduces resource utilization and dimensionality of the problem.

- **Traffic filtering:** Data set was filtered according to the requirements. To create the feature vectors, information from the headers was extracted. The resulting data set was used to calculate statistics and other principle components.
- **Applying machine learning algorithms:** The objective of this step is to find the algorithms that detect the attack effectively. For this purpose, the false positives should be low and detection rates should be higher.

4.10 Attacks detected

There are a plethora of attacks which needs to be detected from the traffic traces. We focussed mainly on four attacks namely:

- Syn Flood Attacks
- Smurf Attack
- Port Scan
- Port Sweep

4.10.1 Syn flood attack

It is a form of DoS attack in which a succession of SYN requests are send by the attacker at the target's system. This is done to make the target system unresponsive to genuine traffic by consuming the server's resources. While making the TCP connection, following steps are followed as shown in Fig:

- The client initiates the connection by sending SYN request to the server.
- The server responds to the request by sending SYN-ACK to the client.
- Finally the connection is made after the client responds with ACK to the server.

The procedure explained is the foundation of every connection which is established through TCP protocol and is called as TCP three-way handshake. Contrary to the normal procedure, in the SYN flood attack, client does not respond to the server with ACK as shown in Figure 4.4.

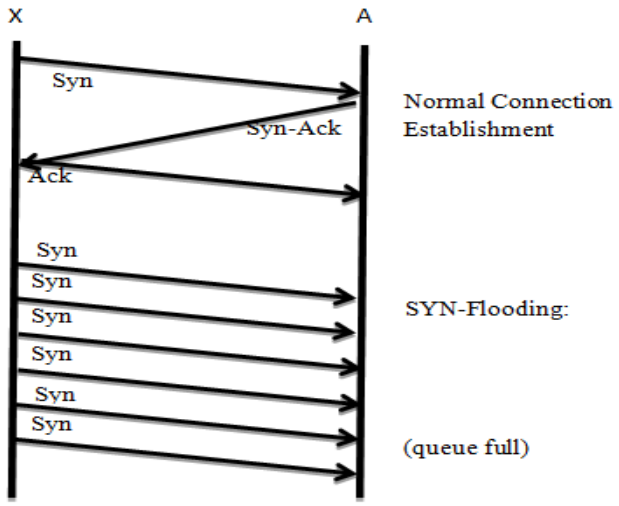

Figure 4.4: Syn flood attack

The server waits for ACK for a particular period of time. Half-open connections exploit the resources of the server and ultimately availability exceeds. Then server is unable to respond to any of the clients, whether malicious or legitimate. This denies services to the genuine clients and in some cases can also system if functions are starved of the resources.

Attributes to be considered while detecting the attack:

We analyzed the TCP syn flood attacks and concluded some of the important attributes to be considered while detection of attack as shown in Table 4.1.

Table 4.1: Syn flood attack attributes

Attributes	Description
Timestamp	Time when the event has occurred
SYN Requests	Request for establishing the connection
Destination IP	IP address of destination host
Traffic Count	Total count of packets while communication
Average packets	Average packets per second
Average packet Size	Average size of the packets in whole traffic

4.10.2 Smurf attack

Smurf attack is also one of the types of DDoS attack. In this enormous amount of ICMP packets are broadcasted to a network with the help of IP broadcast address and

the source address of the packets is altered by victim's spoofed IP. As the request is broadcasted, so all the devices which are active and get the request will respond to the source IP. In this manner, victim's machine gets flooded with the traffic and hence will result in slowing down or crashing the victim's machine.

The effectiveness of the detection process depends on the testing parameters. The planned variables are depicted in the Table 4.2.

Table 4.2: Smurf attack attributes

Attributes	Description
Machines in broadcast segment	Alteration in number of machines available in broadcast segment
Destination IP	IP address of destination host
Protocol	ICMP

4.10.3 Port scan

It is a process in which requests are sent by the client to a number of port addresses on the host. The motive behind doing so is to find the active ports on the host. Although it is not nefarious process in itself but it is exploited by attackers to probe the host machine services. The aim of probing the target machine's services is to exploit the well known vulnerabilities of the active services. However, there are many uses of the port scan which have nothing to do with the attacks. Those uses are just meant to know the services running on the other machine. There is the possibility of port scanning if the flow count will be large, packet size will be small, packet count will be small and destination IP same. The parameters considered are shown in Table 4.3.

Table 4.3: Port scan attributes

Attributes	Description
Flow Count	Flow of traffic
Source Port	It is used for application identification. Although, now-a-days it has become difficult to figure out the application because port numbers are dynamically allocated. Still it proves useful to identify suspected traffic many times.
Packet Size	Size of single packet
Packet Count	Total number of packets in the traffic
Destination IP	IP address of destination host

4.10.4 Port sweep

This is the process of scanning a number of hosts to determine a specific listening port. This is used to find out a particular service on the multiple hosts. Flow count helps to monitor communication patterns. In case of port sweep it will be large and the packet size will be small. Important attributes required to identify port sweep are described in the Table 4.4.

Table 4.4: Port sweep attributes

Attributes	Description
Flow Count	Flow of traffic
Packet Size	Size of single packet
Destination Port	Port number of destination host
Packet Count	Total number of packets in a flow

4.11 Classification

Classification technique was used to measure the results. In classification, unseen instances are classified into various categories with the help of classified examples

presented in the algorithm. Many a times classification is also referred as supervised learning because the outcomes are supervised by the training instances.

In this technique, first of all dataset was pre-processed and feature selection was done as depicted in Figure 4.5. Then classification was done by dividing the whole dataset into subsets. Among that one subset was used to test the model, whereas others were used for the training purpose.

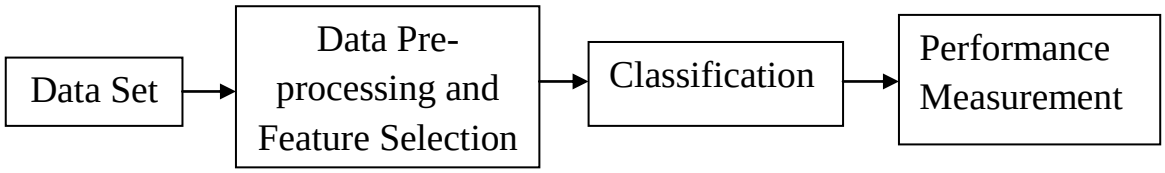

Figure 4.5: Traffic analysis

In the analysis ten subsets were created as we chose 10 fold cross-validation. The whole process was repeated ten times. Every time different subset was used for the testing purpose. Based on the outputs finally performance of the machine learning algorithms on the given dataset was measured.

Four machine learning algorithms were chosen to implement on the dataset, as described below:

4.11.1 Naive Bayes

Naive Bayes is a probabilistic classifier which applies Bayes' theorem with independence speculations between the features. It has been in usage since 1950s and is still popular for text categorization. Based on word frequencies it categorizes the documents into various categories. Naive Bayes classifiers are simple and scalable. Naïve Bayes classifier uses a number of algorithms to train the classifier. All the algorithms used follow the principle that value of one attribute is independent of value of other attribute.

4.11.2 Bayes Net

Bayes Net presents set of random variables and their dependencies via acyclic graph. For an instance, Bayes net which is also known as Bayesian network is used to represent the connection between ailments and symptoms. This helps to detect the probability of diseases provided the symptoms are given. The typical use of Bayes net

is to explain and model a domain. It helps to for decision making under uncertainty and finds strategies for solving tasks.

4.11.3 J48

J48 is a decision tree based on predictive machine learning. It decides the dependent variable on the basis of parameter values of available data. The inside nodes of the tree signifies the various parameters, the branches in between tells the probable values those attributes can have and the terminate nodes indicate the final values also defined as classification. The predicted attribute is called dependent variable. The attributes which helps in the estimation of dependent variables are called as independent variables. In order to classify Decision tree classifier creates decision tree based on available training data.

4.11.4 Random Forest

This algorithm is method for regression, classification, etc and is developed by Adele Cutler and Leo Breiman. It operates by decision tress during the training time and the output of those trees is mean prediction of each tree. They can be used for ranking of importance of variables in natural way. The main drawback of Random Forest algorithm is that it is biased for those attributes which have more levels in the decision tree.

Chapter 5

Results and Discussions

This chapter gives the outputs of analysis done on the captured university traffic dataset by using various machine learning algorithms. The first part describes the results for usage behaviour and the next part gives description of the outputs of suspected malicious activities in the university traffic.

5.1 Usage pattern

5.1.1 Protocol hierarchy

Table 5.1 describes the protocol hierarchy analyzed from the captured university traffic. As it is clearly seen that majority of the traffic was caused by the ARP protocol, closely followed by IPv4. IPv6 and LLC were also responsible for creating some chunk of the traffic.

Most of the traffic was in ARP domain, precisely 47.55%. Lots of ARP traffic may lead to ARP spoofing, in which attacker's MAC address is associated with IP address of other host. This causes the attacker to receive the traffic meant for other host. This is a type of man in the middle attack. As ARP spoofing is only limited to local network, so it is advised to have static ARP entries for gateway to mitigate the attack.

ARP traffic in the table is followed by IPv4 traffic which is further grouped into TCP, UDP, ICMP and IGMP. Traffic caused by the NetBIOS is comparatively more in the TCP as well as UDP domain. NetBIOS gives services related to session layer. It facilitates the communication of applications on different computers over a LAN.

NetBIOS session service allows the computers to create connection and also facilitates error detection and recovery. However, there are a number of vulnerabilities if this port remains open. NetBIOS is bound over TCP/IP which results in glaring security issues. NetBIOS enables the users to share printers and files by making each computer a server or a file. Ultimately, this permits each computer on the LAN to print, share or store files with the help of shared service from other computer. Binding NetBIOS to TCP/IP makes the LAN a part of entire Internet. This

makes any computer on the Internet to print and share files from the computers of private shared LAN.

Table 5.1: Proportion of various protocols in the university traffic

Protocol			Percentage
IPv4			38.77
	TCP		11.70
		DNS	0.14
		NetBIOS Session Service	3.62
		HTTP	0.57
		SSL	0.09
	UDP		12.97
		DNS	3.37
		NetBIOS Datagram Service	0.06
		NetBIOS Name Service	4.37
		Bootstrap Protocol	1.00
	ICMP		7.67
	IGMP		1.01
IPv6			11.80
	UDP		10.60
		DHCPv6	0.22
		DNS	8.44
		HTTP	1.94
	ICMPv6		1.20
LLC			1.61
	STP		0.74
ARP			47.55
Others			0.27

5.1.2 Web categorization

Web Categorization is the assignment of URL's to the predefined categories. We collected the data of network usage within the university for seven days and then filtered the FQDN's from it. After comparing the FQDN's with the databases available on the Internet we found out the results depicted in Table 5.2.

Table 5.2: Web categorization

Category	Users
Shopping	58697
Entertainment	63740
Information Technology	285221
Business and Economy	57263
Advertisement	56728
Social Networking	47343
Sports	26595
News	40636

We analyzed that majority of the users in the university were opening sites related to information technology. Information technology sites contributed 45% of the total share, which seems to be a healthy trend in the university. Hits on the entertainment, business and economy, shopping, advertisement and social networking sites were nearby in percentage as shown in Figure 5.1.

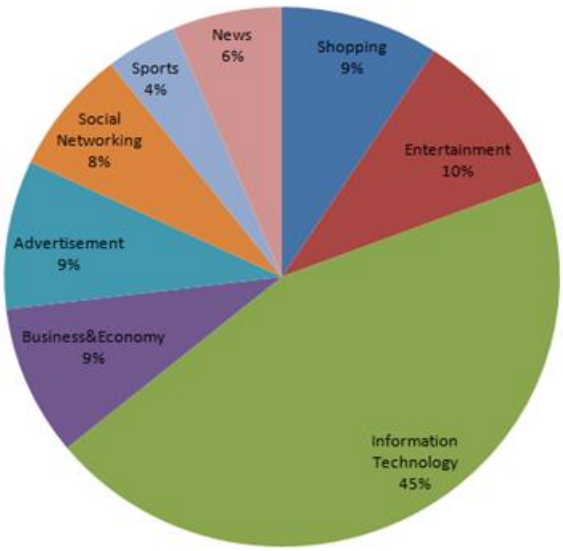

Figure 5.1: Pie chart depicting categorization of traffic

Moving further, sites related to news were opened rarely by the users and the least trend was to open sites of sports category.

5.1.3 Top FTP servers

After analyzing another dataset, we found out the top FTP servers in that particular period of time. FTP has client-server architecture. FTP can run both in passive as well as active mode. The mode tells that how the connection was made between client and server. In both the modes, client creates connection to FTP server port 21 from usually a random source port. In the Table 5.3 we figured out the destination IP addresses having port 21.

Table 5.3: Top FTP servers

Server	Percent
124.40.41.*	11.92%
69.192.2.*	10.37%
150.65.7.*	10.06%
198.63.231.*	9.97%

We figured out the top FTP servers and the percent use of them.

5.1.4 Top attackers

We have analyzed the top attackers from the dataset based on the ICMP protocol. The information depicted in the Table 5.4 can act as a part of compliance report for HIPAA and SOX.

Table 5.4: Top attackers

Attacker	Hits
172.31.149.203	8411
172.31.151.1	8303
172.31.147.204	7865
172.31.147.244	5176
172.31.168.103	4964

HIPAA sets the rules for saving sensitive data of the patients. It has two rules i.e. “Security Rule” and “Privacy Rule” which are the standards for companies to ensure availability, confidentiality and integrity of the patient’s data. On the other hand, SOX is an act passed by United States government to protect general public and shareholders from fraudulent practices and accounting errors. SOX also aim to upgrade accuracy of corporate disclosures. Although we have analyzed these attackers based only on the ICMP protocol yet but it can be extended for better results. The SOX was the response to series of financial scandals occurred in the companies that rattled investor’s confidence. The main motive of SOX was to improve corporate accountability and governance. Now it is mandatory for organizations to comply with SOX.

These acts formalize internal checks with organizations and make sure that reporting exercises full disclosure. They also ensure transparency in corporate governance. Our results along with other information can act as compliance report for HIPPA and SOX.

5.2 Detection of malicious traffic

We analyzed the collected dataset to detect the malicious activities going on within the campus. We analyzed the traffic only for four malicious activities i.e. syn flood attacks, smurf attack, port scan and port sweep. We filtered the traffic according to the parameters required for analysis for various attacks and then applied machine learning algorithms on them.

We used classification algorithms for our analysis. The performance of these algorithms is predicted by the fact that how correctly it classifies the given dataset. In the experiment, we examined the classifiers models with the help of k-fold cross-validation mode. It is widely acknowledged testing procedure in which dataset is segmented into k disjoint segments. After that data mining algorithm is trained with the help of k-1 segment and the other segments are used for testing purpose. This whole method is repeated k times and at last the results recorded each tie are averaged. In our thesis, we chose value of k to be 10 i.e. we predicted the error rate of various classifiers with the help of 10 fold cross-validation.

TP Rate: It describes the proportion of instances anticipated as positive which were actually positive.

FP Rate: It describes the proportion of instances anticipated as positive which were actually negative.

Precision: It is described as the division of occurrences that truly classified in a class and the total occurrences classified in the class.

$$\text{Precision} = \frac{\text{True Positive}}{(\text{True Positive+False Positive})}$$

Recall: It is defined as the ratio of occurrences classified in a class to that of actual total occurrences in that class.

F-Measure: It is the measure for recall and precision and is calculated as:

$$\text{F} - \text{Measure} = 2 \times \frac{(\text{Precision} \times \text{Recall})}{(\text{Precision} + \text{Recall})}$$

5.2.1 Syn flood attack

First of all, we detected the syn flood attack which is a type of DoS attacks. We calculated the correctly classified instances and incorrectly classified instances in Table 5.5. Naïve Bayes classified the instance correctly by 98.1632% and J48 gave the worst performance comparatively.

Table 5.5: Syn flood classification

Machine Learning Algorithm	Correctly Classified Instances	Incorrectly Classified Instances
Naïve Bayes	98.1632%	1.8368%
Bayes Net	98.0514%	1.9486%
J48	96.2306%	3.7694%
Random Forest	97.8757%	2.1243%

Attacks were detected with the help of four machine learning algorithms namely Naïve Bayes, Bayes net, J48 and Random Forest. Further TP Rate, FP Rate, Precision, Recall and F-Measure were measured as shown in Table 5.6.

Table 5.6: Comparative study of machine learning algorithms for syn flood attack

Machine Learning Algorithm	TP Rate	FP Rate	Precision	Recall	F-Measure
Naïve Bayes	0.982	0.009	0.984	0.982	0.982
Bayes Net	0.981	0.011	0.982	0.981	0.981
J48	0.962	0.026	0.965	0.962	0.963
Random Forest	0.979	0.014	0.98	0.979	0.979

Then top syn flooders were detected within the campus and we came out with the IPs mentioned in Table 5.7. Traffic was filtered according to the attack which needed to be detected.

Table 5.7: Top syn flooders

IP Addresses
172.31.121.240
172.31.13.19
172.31.146.66
172.31.160.202
172.31.17.105
172.31.17.128
172.31.5.175

For detecting syn flood attacks mainly six attributes were considered. Those six attributes were time of occurrence of event, syn request field, traffic count, average number of packets, destination IP and average packet size. On the basis of these parameters results were analyzed.

5.2.2 Smurf attack

Second attack detected shown in Table 5.8 was also a type of DoS attack known as smurf attack. In this also better results were given by Naïve Bayes machine learning algorithm.

Table 5.8: Smurf attack classification

Machine Learning Algorithm	Correctly Classified Instances	Incorrectly Classified Instances
Naïve Bayes	99.3372%	0.6628%
Bayes Net	98.858%	1.142%
J48	97.8678%	2.1322%
Random Forest	99.2094%	0.7906%

Further true positives and true negatives were also calculated as depicted in Table 5.9. The parameters considered for finding smurf attack were number of machines in the broadcast segment, protocol should be ICMP and destination IP was also checked.

Table 5.9: Comparative study of machine learning algorithms for smurf attack

Machine Learning Algorithm	TP Rate	FP Rate	Precision	Recall	F-Measure
Naïve Bayes	0.993	0.003	0.994	0.993	0.993
Bayes Net	0.989	0.002	0.991	0.989	0.989
J48	0.979	0.01	0.981	0.979	0.979
Random Forest	0.992	0.005	0.992	0.992	0.992

5.2.3 Port scan

Next category of attacks detected was probing. In case of detecting port scanning, destination IP was scrutinized. Other parameters considered were packet size, source port and packet count. It is seen in Table 5.10, Naïve Bayes correctly classified 98.75% instances which is relatively more than Bayes net, J48 and random forest as well.

Table 5.10: Port scan classification

Machine Learning Algorithm	Correctly Classified Instances	Incorrectly Classified Instances
Naïve Bayes	98.75%	1.25%
Bayes Net	94.25%	5.75%
J48	96.5%	3.5%
Random Forest	96.75%	3.25%

Bayes net classified the instances least correctly i.e. 94.25% which is approximately 4% less than that of Naïve Bayes. Table 5.11 depicts that true positive rate of Naïve Bayes was maximum i.e. 0.988 with least false positive rate of 0.003.

Table 5.11: Comparative study for machine learning algorithms for port scan

Machine Learning Algorithm	TP Rate	FP Rate	Precision	Recall	F-Measure
Naïve Bayes	0.988	0.003	0.988	0.988	0.988
Bayes Net	0.943	0.114	0.943	0.943	0.943
J48	0.965	0.063	0.965	0.965	0.965
Random Forest	0.968	0.054	0.968	0.968	0.968

5.2.4 Port sweep

Port Sweep is similar as network scanning. Network is scanned to see whether a particular type of port is open or not on the hosts. Table 5.12 illustrates that Naïve Bayes and J48 algorithms give same and very high accuracy i.e. 99.5% in detecting port sweeps.

Table 5.12: Port Sweep Classification

Machine Learning Algorithm	Correctly Classified Instances	Incorrectly Classified Instances
Naïve Bayes	99.5%	0.5%
Bayes Net	99%	1%
J48	99.5%	0.5%
Random Forest	99%	1%

Attributes considered for detecting poet sweeps in traffic were flow count, destination port and packet count as well as packet size. Although, all the machine learning algorithms gave almost same results in analyzing port sweep but Naïve Bayes and J48 were somewhat better as shown in Table 5.13.

Table 5.13: Comparative study of machine learning algorithms for port sweep

Machine Learning Algorithm	TP Rate	FP Rate	Precision	Recall	F-Measure
Naïve Bayes	0.995	0.001	0.995	0.995	0.995
Bayes Net	0.99	0.003	0.99	0.99	0.99
J48	0.995	0.001	0.995	0.995	0.995
Random Forest	0.99	0.003	0.99	0.99	0.99

It is crystal clear from the all above results that Naïve Bayes classifier is better for analyzing malicious traffic. Naïve Bayes classifier outperformed other algorithms like random forests because of some sound theoretical reasons. Naïve Bayes also needs fewer amounts of data for training to predict attributes for classification, which is a great advantage of it over other classifiers. Naïve Bayes gave better results for our large dataset because it does not have complicated iterative attribute estimation.

While classifying an instance from dataset, Naïve Bayes classifier considers each parameter of the dataset separately. It assumes that every attribute works independently and hence gives the results accordingly. In the results, it is vivid that Naïve Bayes worked well as it is less sensitive to the irrelevant attributes. Many a times attributes are not independent of each other still it works well because dependencies scatter evenly or cancels out. Also the optimal classifier is obtained as long as both the estimated and actual distributions agree on the most-probable class and not necessarily dependent upon appropriateness of independent assumption.

Chapter 6

Conclusion and Future Scope

To our knowledge, a lot of research has been done to analyze Internet traffic in general but nothing much is done particularly on the university traffic. Hence, we have worked on this area considering decentralized network of universities which makes them more prone to attacks. As a predominant contribution of this paper, we have shown that there are undesirable characteristics prevailing in the network traffic. We have shown a systematic approach to identify unwanted network traffic patterns giving the evidences of DDoS attacks. Our results include insights of the traffic patterns from where we have deduced the threats to the university network. The outcomes of our paper are also giving a good view on how the university network is being utilized by the users.

There are further enhancements which can be done in this thesis. Instead of storing and then analyzing the traffic, active monitoring can be done. Properties of the packets can be inferred by continuously sending them across the network. By analyzing the live traffic infrastructure and other resources can be saved from the detrimental effects as immediately countermeasures can be taken. Further research can also be done on figuring out the source of malicious activity. As the origin of malign activity can vary over time, so some dynamic approach can be helpful to inspect the malicious behaviour.

References

[1] Y. Zhang, "Residential network traffic and user behavior analysis," M.S. thesis, KTH Information and Communication Technology, Stockholm, Sweden, 2010, pp. 276.

[2] S. William. Stewart (2014). BBC report on Internet shopping. [Online]. Available: http://news.bbc.co.uk/1/hi/business/4630472.stm

[3] P. Sandford, "Inferring malicious network events in commercial ISP networks using traffic summarisation," Ph.D. dissertation, Loughborough Univ., UK, 2012.

[4] S. Pereira, J. Maia and J. Silva, "ITCM: A Real Time Internet Traffic Classifier Monitor," International Journal of Computer Science & Information Technology, vol. 6, 2015.

[5] B. Stewart. (2015). Internet History. [Online]. Available: http://www.livinginternet.com/i/ii_summary.htm

[6] R. Wigner. (2015). Internet World Stats. [Online]. Available: http://www.internetworldstats.com/stats.htm

[7] S. Venkataraman, "Traffic Analysis for Network Security using Learning Theory and Streaming Algorithms," Ph.D. dissertation, School of Computer Science, Carnegie Mellon Univ., Pittsburgh, 2008.

[8] D. Barbara, S. Jajodia, and N. Wu, "AD AM: Detecting intrusions by data mining," in IEEE Workshop on Information Assurance and Security, 2001.

[9] A. Lakhina, M. Crovella and C. Diot, "Mining anomalies using traffic feature distributions," in ACM SIGCOMM Computer Communication Review, Pennsylvania, 2005, pp. 217-228.

[10] B. Eshete, "Effective Analysis, Characterization, and Detection of Malicious Activities on the Web," Ph.D. disseration, Univ. of Trendo, Italy, 2013.

[11] G. Egan, et al., "Symantec internet security threat report trends for 2010," Symantec, 2010.

[12] C. Grier, et al., "Manufacturing compromise: the emergence of exploit-as-a-service," in *Computer and communications security*, 2012, pp. 821-832.

[13] K. Lan and J. Heidemann, "On the correlation of internet flow characteristics," Univ. of South California, CA, 2003.

[14] D. Zuev and A. Moore, "Traffic classification using a statistical approach," in Passive and Active Network Measurement, Springer Berlin Heidelberg, 2005, pp. 321-324.

[15] A. Lakhina, M. Crovella and C. Diot, "Characterization of network-wide anomalies in traffic flows," in Proceedings of the 4th ACM SIGCOMM conference on Internet measurement, ACM, 2004, pp. 201-206.

[16] K. Xu, Z. Zhang and S. Bhattacharyya, "Internet traffic behavior profiling for network security monitoring," Networking, IEEE/ACM Transactions, vol. no. 16, 2008, pp. 1241-1252.

[17] M. Mahoney, "Network traffic anomaly detection based on packet bytes," in Proceedings of the 2003 ACM symposium on Applied computing, ACM, 2003, pp. 346-350.

[18] M. Kim, Y. Won and J. Hong, "Characteristic analysis of internet traffic from the perspective of flows," *Computer Communications* 29, vol. no. 10, 2006, pp. 1639-1652.

[19] P. Barford and D. Plonka, "Characteristics of network traffic flow anomalies," in Proceedings of the 1st ACM SIGCOMM Workshop on Internet Measurement, ACM, San Francisco, 2001, pp. 69-73.

[20] A. Taylor and B. Rothke, "Network Security Foundations," in Network security: The complete reference, McGraw-Hill, 2003.

[21] V. Paxson, "Bro: a system for detecting network intruders in real-time," Computer networks, vol. no. 23, 1999, pp. 2435-2463.

[22] S. Klerer, "The OSI management architecture: an overview," Network, IEEE 2, vol. no. 2, 1988, pp. 20-29.

[23] N. Weaver, et al., "A taxonomy of computer worms," in Proceedings of the 2003 ACM workshop on Rapid malcode, ACM, 2003, pp. 11-18.

[24] R. Bouckaert, et al., "WEKA Manual for Version 3-7-8," 2013.

[25] M. Dabrowski, et al., "Characteristics and responsibilities involved in a Phishing attack," in Proceedings of the 4th international symposium on Information and communication technologies, Dublin, 2005, pp. 249-254.

[26] C. Papadopoulos, et al., "A framework for classifying denial of service attacks," in Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications, ACM, 2003, pp. 99-110.

[27] H. Spafford, et al., "The Internet worm program: An analysis," ACM SIGCOMM Computer Communication Review 19, vol. 1, 1989, pp. 17-57.

[28] P. Borgnat, et al., "Seven years and one day: Sketching the evolution of internet traffic," In INFOCOM 2009, IEEE, 2009, pp. 711-719.

[29] L. Sony, "Traffic data repository at the WIDE project," in Proceedings of USENIX 2000 Annual Technical Conference: FREENIX Track, 2000, pp. 263-270.

[30] A. Bestavros and M. Crovellaand, "Self-similarity in worldwide web traffic: Evidence and possible causes," Performance Evaluation Review 24, vol. 5, 1996, pp. 160-169.

[31] J. Färber, S. Bodamer, and J. Charzinski, "Measurement and modelling of Internet traffic at access networks," in Proceedings of the EUNICE, vol. 98, 1998, pp. 196-203.

[32] P. Barford, et al., "A signal analysis of network traffic anomalies," in Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurment, ACM, 2002, pp. 71-82.

[33] J. Erman, M. Arlitt, and A. Mahanti, "Traffic classification using clustering algorithms," in Proceedings of the 2006 SIGCOMM workshop on Mining network data, ACM, 2006, pp. 281-286.

[34] A. Moore and D. Zuev, "Internet traffic classification using bayesian analysis techniques," in ACM SIGMETRICS Performance Evaluation Review, ACM, vol. 1, 2005, pp. 50-60.

[35] T. Karagiannis, K. Papagiannaki, and M. Faloutsos, "BLINC: multilevel traffic classification in the dark," in ACM SIGCOMM Computer Communication Review, ACM, vol. 4, 2005, pp. 229-240.

[36] P. Kumar, P. Senthil and S. Arumugam, "Establishing a valuable method of packet capture and packet analyzer tools in firewall," International Journal of Research Studies in Computing, vol. 1, 2011.

[37] M. Kihl, "Traffic analysis and characterization of Internet user behaviour," in Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), IEEE, 2010, pp. 224-231.

[38] H. Abrahamsson, "Internet traffic management," Ph.D. disseration, Mälardalen Univ., Sweden, 2008.

[39] G. Dias et al., “A network security monitor,” in Research in Security and Privacy, Proceedings, IEEE Computer Society Symposium, 1990, pp. 296-304.

[40] R. Nelson, et al., "Analysis of long duration traces," ACM SIGCOMM Computer Communication Review 35, vol. 1, 2005, pp. 45-52.

[41] J. Hong, et al., "A flow-based method for abnormal network traffic detection," in Network operations and management symposium, IEEE/IFIP, vol. 1, 2004, pp. 599-612.

[42] C. Douligeris and A. Mitrokotsa, "DDoS attacks and defense mechanisms: classification and state-of-the-art," Computer Networks, vol. 5, 2004, pp. 643-666.

[43] F. Lau, et al., "Distributed denial of service attacks," in Systems, Man, and Cybernetics, 2000 IEEE International Conference, IEEE, vol. 3, 2000, pp. 2275-2280.

[44] J. Wolfgang, S. Tafvelin and T. Olovsson, "Trends and differences in connection-behavior within classes of internet backbone traffic," in Passive and Active Network Measurement, Springer Berlin Heidelberg, 2008, pp. 192-201.

[45] C. Xiang and S. Lim, "Design of multiple-level hybrid classifier for intrusion detection system," in Machine Learning for Signal Processing, IEEE Workshop, 2005, pp. 117-122.

[46] S. Mitropoulos, D. Patsos, and C. Douligeris, "Network forensics: towards a classification of traceback mechanisms," in Security and Privacy for Emerging Areas in Communication Networks, Workshop of the 1st International Conference, IEEE, 2005, pp. 9-16.

[47] Request for comments (rfc) pages [Online]. Available: http://www.ietf.org/rfc.html

[48] N. Meghanathan, A. Reddy and L. Moore, "Tools and techniques for network forensics," International Journal of Network Security & Its Applications, 2010, pp. 14-25.

[49] K. Kent, et al., "Guide to integrating forensic techniques into incident response," NIST Special Publication, 2006, pp. 800-86.

[50] V. Kumar, H. Chauhan, and D. Panwar, "K-Means Clustering Approach to Analyze NSL-KDD Intrusion Detection Dataset," International Journal of Soft Computing and Engineering, 2013, pp. 2231-2307.

[51] A. McGregor, et al., "Flow clustering using machine learning techniques," in Passive and Active Network Measurement, Springer Berlin Heidelberg, 2004, pp. 205-214.

[52] S. Zander, T. Nguyen and G. Armitage, "Automated traffic classification and application identification using machine learning," in Local Computer Networks, IEEE, 2005, pp. 250-257.

[53] S. Adibi, "Traffic Classification–Packet-, Flow-, and Application-based Approaches," International Journal of Advanced Computer Science and Applications, 2010, pp. 6-15.

[54] F. D. Smith, et al., "Statistical clustering of internet communication patterns," Computing science and statistics, 2003.

[55] R. Braden, "TA pseudo-machine for packet monitoring and statistics," in Proc. of SIGCOMM, vol. 88, 1988.

[56] C. Estan, S. Savage, and G. Varghese, "Automatically inferring patterns of resource consumption in network traffic," in Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications, ACM , 2003, pp. 137-148.

[57] N. Marques, et al., "A characterization of broadband user behavior and their e-business activities," ACM SIGMETRICS Performance Evaluation Review 32, vol. 3, 2004, pp. 3-13.

[58] D. Manners, "The user agent field: Analyzing and detecting the abnormal or malicious in your organization," SANS Institute reading room site, 2012.

Communicated:

H. K. Gill and M. Singh, "Capture University Network Traffic to Analyze Usage Behaviour and to Detect Malicious Activities," in International Symposium on Advanced Computing and Communication, IEEE, 2015.

Video Link

This is link to my YouTube video where I have presented brief summary of my thesis topic:

https://www.youtube.com/watch?v=3JuErjUoHjQ&feature=youtu.be

Turnitin Originality Report

CAPTURE, ANALYZE AND DETECT MALICIOUS ACTIVITIES IN A UNIVERSITY NETWORK TRAFFIC by Harleen Kaur Gill

From Thesis (ME 2013-2015 Batch)

- Processed on 2015年07月15日 00:22 IST
- ID: 555718818
- Word Count: 12386

Similarity Index
4%
Similarity by Source

Internet Sources:
2%
Publications:
3%
Student Papers:
0%

sources:

1 1% match (Internet from 15-Jun-2015)
http://users.cs.cf.ac.uk/O.F.Rana/Antonia.J.Jones/Theses/JRRabaiottiThesis.pdf

2 < 1% match (publications)
Kim, M.S.. "Characteristic analysis of internet traffic from the perspective of flows", Computer