Kingdom of Saudi Arabia
The Long-Term Comprehensive National
Plan for Science,Technology and Innovation
General Secretariat

مدينة الملك عبدالعزيز
للعلوم والتقنية KACST

Project # 08-INF101-4

FINAL REPORT

SAUDI HONEYNET PROJECT

مشروع شبكات العسل السعودية

Principal Investigator, ***Dr. Mohammed Houssaini Sqalli,*** ***Assistant Professor***

Computer Engineering Department

Date: Dhul-Hijjah 1432
Date: November 2011

NATIONAL SCIENCE, TECHNOLOGY & INNOVATION PLAN UNIT
KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

PROJECT INFORMATION

Project #	08-INF101-4
Project Title	Saudi Honeynet Project
Principal Investigator	Dr. Mohammed Houssaini Sqalli
Institution	King Fahd University of Petroleum & Minerals
Strategic Technology Area/Track/Sub-Track	Information Technology / Computer Systems and Networks / IT security and privacy
Award Period (Start Month/Year – End Month/Year)	09/2009 - 08/2011
Extensions (if any)	24 months

Research Team	Senior Personnel				
	No.	Name	Rank	Role	Area of Specialization
	1	Dr. Mohammed H. Sqalli	Assistant Professor	P I	Computer Networks & Security
	2	Dr. Khaled Salah (Sep. 2009-Aug. 2010)	Associate Professor	CO- I	Computer Networks & Security
	3	Dr. Marwan Abu-Amara	Assistant Professor	CO- I	Computer Networks
	4	Dr. Zubair Baig	Assistant Professor	CO- I	Computer Networks & Security
	5	Dr. Farag Azzedin	Associate Professor	CO- I	Computer Networks
	6	Dr. Talal Alkharobi	Assistant Professor	CO- I	Computer Security
	7	Mr. Hakim Adiche	Lecturer	CO- I	Computer Networks
	Other Personnel				
	8	Mr. Yusuf Sharif Hassan		Engineer	
	9	Mr. Virray Ferdinand		Lab Technician	
	Consultant				
	10	Dr. Khaled Salah (Sep. 2010-Aug. 2011), U.A.E.			
	11	Mr. Mahmud Bin Ab Rahman, Malaysia			

المُلخص:

وضع مشروع شبكة العسل السعودي بنجاح نموذجاً يحتذى في المملكة العربية السعودية كلها لجمع وتبادل وتحليل وتحديد التهديدات الأمنية والأنشطة المتعلقة بشبكات الحاسب الآلي والقطاعات الفردية والخاصة والصناعية والحكومية. سيساعد هذا المشروع بدون شك في تعزيز الأمن المعلوماتي في المملكة العربية السعودية، والذي يعد أحد الأهداف الرئيسية التي حددتها الخطة الوطنية للعلوم والتقنية. يضع هذا المشروع نموذجاً عملياً يتألف من "شبكة عسل" بنيت داخل حرم جامعة الملك فهد للبترول والمعادن لجمع البيانات حول سلوك المهاجمين وتحليل وتقييم أمن شبكات الحاسب لدينا. كما يمكن تطوير نموذج مماثل ليتم استخدامه على مستوى المملكة. المشروع كان رائداً أيضاً في اقتراح وتطوير تقنيات جديدة لتحليل سيل المعلومات في شبكة العسل، وبالتالي أتمتة وتبسيط بعض المهام ذات الصلة لتحليل هذه المعلومات. نتج عن هذا البحث عدد من المواضيع العلمية التي تم نشرها في مؤتمرات علمية معتبرة. الأمر الأكثر أهمية حالياً، هو استخدام شبكة العسل السعودية كأداة أمنية هامة حيث يتم إعداد تقارير أمنية أسبوعية تلخص الأنشطة المؤذية والمشبوهة التي تستهدف شبكات الحاسب الآلي بجامعة الملك فهد للبترول والمعادن. كما يتم إرسال هذه التقارير للفريق السعودي للاستجابة لطوارئ الحاسبات السعودية (CERT-SA) والذي يعتبر الجهة الحكومية المسؤولة عن مراقبة أمن الشبكات في المملكة العربية السعودية. بالإضافة إلى ذلك، تم تقييم عمل هذا المشروع من قبل المنظمة الدولية لمشروع شبكة العسل، وقد تأهل مشروع شبكة العسل السعودية بالجامعة لإنشاء الفرع السعودي ضمن المنظمة الدولية لمشروع شبكة العسل كأول فرع في المملكة. إنشاء الفرع السعودي لشبكة العسل أتاح لنا موارد ثمينة وتواصل أفضل مع الباحثين في مجال أمن المعلومات. وقد شارك عضوين من فريق المشروع في ورشة العمل السنوية لمشروع شبكة العسل، كما زار عدد من أعضاء فريق المشروع فرعين مهمين لشبكات العسل بدول أخرى. وقد أسفرت هذه الزيارات والتواصل مع الباحثين عن عدد من المشاريع البحثية التي سيتم طرحها في المستقبل القريب.

SUMMARY

The Saudi Honeynet Project (SAHNET) has successfully set a model to follow in the whole Kingdom of Saudi Arabia to collect, share, analyze, and identify security threats and activities pertaining to computer networks of individual, private, industrial, and government sectors. The project will with no doubt help enhancing the IT security in KSA, which is one of the primary goals set by the NSTIP strategic plan. The project has set a practical model comprised of a Honeynet network built within the KFUPM campus for collecting data about attackers' behavior as well as analyzing and assessing the security of our networks. A similar model can be extended to be deployed at the level of KSA. The project has also pioneered in proposing and developing new techniques for the analysis of the Honeynet traffic, and thus automating and simplifying some of the tasks related to analyzing Honeynet traffic. Based on our research work, a number of publications have been produced and presented in reputable conferences. More importantly, and currently, as a way of disseminating extremely useful information to users, the SAHNET project generates weekly different types of security reports that summarize malicious and suspicious activities targeting KFUPM networks. These reports are shared with the Saudi Computer Emergency Response Team (CERT-SA) which is the government organization responsible for monitoring the security of the KSA networks. In addition, the work of this project has been evaluated by the international organization of Honeynet Project, and has qualified us to establish our own Saudi Chapter for the first time ever. Establishing a Saudi Chapter has exposed us to valuable resources and researchers in the field of IT security. Our researchers have participated in the annual HP workshop and have visited two major Honeynet Chapters. Such visits and interactions have resulted in a number of future research projects which will be proposed in the near future.

ACKNOWLEGEMENT

The investigators acknowledge the support provided by King Abdulaziz City for Science and Technology (KACST) through the National Science & Technology Unit at King Fahd University of Petroleum & Minerals (KFUPM) for funding this work under project # 08-INF101-4 as part of the National Science, Technology and Innovation Plan.

TABLE OF CONTENTS

Section/Details Page

LIST OF FIGURES

LIST OF TABLES

1.0 INTRODUCTION

The number of Internet users in the Kingdom of Saudi Arabia (KSA) has increased drastically over the past decade from 200,000 users in December 2000 to 11,400,000 users in December 2010 which represents a 43.6% penetration; as per the Communications and Information Technology Commission (CITC) [1]. In addition, businesses and governments use the Internet to provide vital and critical information to their clients and to the world at large, and they are increasingly using the Internet to replace manual methods of collecting information and providing government services. The e-government program known as *Yesser* was established in KSA in 2005 with the objective of digitizing government services and transactions [2]. The current status of the Saudi e-government program provides clear evidence of the rapid progress in this area as an increasing number of government agencies are transforming their services from traditional delivery modes to e-services. This transformation requires a thorough understanding of the implication of such changes, including the security issues.

Computer network security is a major area of concern for a diverse range of people from normal home users to businesses trying to protect their resources from unauthorized access. The moment a computer is connected to the Internet, it is physically connected to millions of other computers in the network. There is a constant threat from malicious users who are trying to disrupt normal operations or to steal sensitive or proprietary information. Network security is a prominent feature of the network, ensuring accountability, confidentiality, integrity, and above all protection against many external and internal threats such as hacking, denial of service attacks, worms, Trojans, etc., that may arise from both local as well as global networks such as the Internet.

One of the biggest obstructions to the benefits that the Internet and the global information society can provide is disruption by Internet-based cyber-attacks [3]. Cyber-attacks include network attacks against vulnerable services, attacks on computer applications, and intrusions that are typically attempted from outside the organization with the intention of crashing the network. They could also include hijacking the computing power, stealing confidential information such as credit card numbers from the network, or using a comprised machine to launch further compromises. Furthermore, cyber attackers use a group of compromised computers controlled remotely (i.e., through botnets) to spread worms, Trojan horses, or backdoors throughout the global information society. Several botnets have been identified and removed from the Internet. For example, the Dutch police found a 1.5 million node botnet [4], and the Norwegian ISP Telenor disbanded a 10,000-node botnet [5, 6]. It has been estimated that up to one quarter of all personal computers connected to the Internet may be part of a botnet [7].

Another important element of security is to understand the malicious techniques and tactics of attackers. Capturing such malicious activity allows for studying and understanding the operations and motivation of attackers, and subsequently helps to enhance security of computers and networks. Honeynets have recently gained a considerable amount of interest as a proactive system to diffuse hostile activities in a network. A key feature of Honeynets is the ability to attract, control, and monitor activities of cyber attackers. A Honeynet is a network designed to gather information on security threats, and it can be used by organizations to proactively improve their network security. A Honeynet can be used to assist system administrators in identifying malicious traffic in the enterprise network. By its very nature, a Honeynet has no production value and should not be generating or receiving any network traffic. Any traffic to or from the Honeynet is assumed to be suspicious in nature. The key requirements to successfully implement a Honeynet are: data control, data capture, and data analysis [8]. The Honeynet is an effective concept that can be used to understand the threats that exist in the networks. It provides tools such as Honeywall and other data capture and data analysis tools to learn about the vulnerabilities in networks. The Honeynet architecture comprises of a diverse set of Honeypots and several other types of tools. A Honeypot has been defined as a security resource whose value lies in being probed, attacked, or compromised [9].

The main goal of the Saudi Honeynet Project (SAHNET) was to lay the ground for a platform that provides information surrounding security threats and vulnerabilities currently active in the networks of the Kingdom of Saudi Arabia. In addition, the project aimed at sharing the findings with the public and the wider IT community. One of the main achievements of the project is that it has indeed led to the establishment of a Honeynet lab within the KFUPM campus for regularly collecting data on attacker behavior. Therefore, different Honeynet elements have been deployed on the KFUPM campus at multiple major sites; for the purpose of collecting, analyzing, and assessing the security of our networks. This deployment includes a central site where all the data is gathered, in addition to multiple sensors across the campus where locally observed data is collected. The Honeynet deployed at KFUPM also includes one sensor that is operational in the perimeter of KFUPM, i.e., the traffic is not filtered by the firewall, and this allows the detection of attacks that could be targeting many other organizations in the Kingdom at large.

Therefore, it has become possible to identify and report vital information and statistics on existing and new malwares as well as other types of attacks that are currently targeting the KSA Internet. The Honeynet deployed at KFUPM is currently being used to collect valuable information about attackers. We have used a diverse set of tools for the collection, detection, and analysis of the Honeynet data, and to identify infected and compromised machines. The very architecture used at the level of KFUPM can be extended to a larger and more complex Honeynet to be deployed at different locations in the Kingdom's networks, and discussion about this issue has already been initiated with the Saudi Computer Emergency Response Team (CERT-SA), an organization with the authority to lead such an effort. CERT-SA is the government organization responsible for monitoring the security of the KSA networks as well as providing awareness and issuing alerts and advisories related to cyber-security to the KSA community at large.

Another major contribution of the project is in proposing and developing new techniques for the analysis of the Honeynet traffic, and thus automating and simplifying some of the tasks related to analyzing Honeynet traffic. The current Honeynet deployments do not include anomaly detection schemes to identify anomalies in the Honeynet traffic. Anomaly detection is useful for detecting zero day attacks and unknown attacks in the network. A Honeynet also collects a substantial amount of data and any incoming data to the Honeynet is considered malicious. Many Honeynet deployments currently use Snort, a Signature-based Intrusion Detection tool, to detect malicious activities; but it is known to generate high rate of false positives [10]. One of the main contributions of the work in this project is to evaluate different candidate features and use the best ones and their corresponding threshold levels to classify the different malicious activities or anomalies seen in Honeynets. Based on this research work, a number of publications have been produced and presented in reputable conferences. In addition, a number of graduate students or research assistants have also participated under the supervision of the principal investigator and co-investigators, working on either the research component or the deployment part of this project.

The data collected by SAHNET includes information such as malware downloads, scanning, and other malicious activities. In addition, a post-analysis of the malwares detected is performed using well known sandboxes, with generation of weekly reports to summarize all the findings including attackers' activities. These reports are also shared with CERT-SA for the purpose of harnessing our local observations and findings with global reports generated at CERT-SA, to improve network security of the entire Kingdom and also to make the public aware of the types of attacks that are targeting KSA. To the best of our knowledge, this is the first time that such data has been collected in KSA. Similar data is collected worldwide by organizations such as ShadowServer and are shared with other organizations worldwide, including CERT-SA. However, the reports provided by such organizations provide information about the attacks initiated from KSA and targeting other countries, rather than attacks that may be perpetrated against the Kingdom.

The SAHNET project provides a plethora of information about attacks initiated from Saudi Arabia or from outside and targeting the Saudi networks; thus providing an additional level of information that was missing. CERT-SA is the government body that has the authority to make use of the information collected. Therefore, we are continuously collaborating with CERT-SA for the purpose of making good use of the reports generated in the SAHNET lab at KFUPM. Based on these reports, CERT-SA will also have the authority to communicate with other worldwide organizations and ISPs to notify them about any illegitimate activities that are initiated from their sites and targeting KSA. CERT-SA also represents the interface to other KSA government organizations and the corporate sector; therefore we believe that through the outcomes of the Saudi Honeynet Project, we have contributed to the improvement of cyber-security in KSA.

As part of this NSTIP project, we have also joined the Honeynet Project Organization (HP) as a chapter in July 2010 after fulfilling all the stipulated requirements. This representation provided us with cutting-edge information about the tools and technologies being used by the HP. In addition, this enabled our team to collaborate with the global Honeynet Project community researchers who are involved in investigating IT security on the global scale. We have also participated in the annual HP workshop and visited two major Honeynet chapters, which has led to some collaboration agreements with them. In addition, we had visits to KFUPM from experts within the HP who delivered security workshops and training for the KFUPM community. Three conferences have been attended by our team members where three papers that were the submitted as part of the outcomes of this project, were presented.

Finally, valuable findings of this project have also been disseminated to CERT-SA, which represents the interface to other KSA government organizations and the corporate sector; thus we believe that we have contributed to the improvement of cyber-security in KSA. The project also increased the awareness of cyber-attacks in KSA based on the data collected and the reports generated. In addition, the SAHNET team is currently responsible for translating the OUCH! newsletter into Arabic, and thus serving the wider IT Arab community. OUCH!, is the free, monthly security awareness newsletter, issued by SANS, renown as one of the most trusted and largest source for computer, network, and information security training and research in the world.

In summary, the proposed project has contributed to achieving the objectives of the national plan of science and technology by providing an infrastructure to better understand the weaknesses of the local networks and then secure and protect them from malicious attacks. The outcomes of this project were manifold. First, the project provided better understanding of different threats and vulnerabilities that can affect the Internet in Saudi Arabia. Second, it helped us in building local expertise and knowledge-base of the trend of cyber threats and corresponding countermeasures. Third, it allowed the dissemination of valuable findings of this project among KSA government organizations and corporate sector; thus contributing to improving confidence in cyber information security. Fourth, it increased the awareness of cyber-attacks in KSA. Fifth, the research and experimental findings carried out in this project were used to establish a Honeynet lab which is being used as part of teaching undergraduate and graduate courses related to cyber-security.

2.0 OBJECTIVES

The ultimate objective of this project was to setup and implement a prototype of a Honeynet to be deployed on the KFUPM campus networks to collect, analyze, and assess the health and security of these networks. This pilot Honeynet can then be extended to a larger and more complex Honeynet to be deployed at different locations in KSA networks.

The primary objectives of the project, which we believe have all been achieved, can be summarized as follows:

1. In-depth exploration of the best practices to design, test, analyze, and implement a Honeynet.
2. Design and deploy a pilot Honeynet on the KFUPM campus networks to carry out experimentations and evaluate their performance.
3. Design the Saudi Honeynet Project and provide recommendations for the deployment of Honeynets kingdom wise at different locations in KSA networks.
4. Build local expertise and knowledge-base in installing, integrating, and developing Honeynets in KSA.
5. Use the research and experimental findings carried out in this project to establish KFUPM undergraduate and graduate labs to teach cyber-attacks and countermeasure mechanisms.
6. Disseminate valuable findings of this project among KSA government organizations and corporate sector thus contributing to improve confidence in the security of KSA networks.
7. Develop a center of excellence or consultancy for other local government and private organizations for the research and development as well as deployment of Honeynets in KSA.

3.0 LITERATURE REVIEW

3.1. Honeynets

Honeynets are constituted of several clients, called Honeypots, which are responsible for detecting hacker activity within a network. There are two primary categories of Honeypots, *research* and *production*. A research Honeypot [11] is deployed to collect information on new attacks and the blackhat community. Usually, detecting an attack is a strenuous and time-consuming process. However, with a Honeypot, which is essentially insignificant in actual data processing, any activity on the Honeypot machine is immediately flagged to be malicious. Therefore, if an administrator observes traffic flowing into a Honeypot, he can initiate a traffic analysis procedure to study the purpose of such traffic. If the traffic is deemed to be malicious, the administrator must then decide when enough data has been collected and the Honeypot needs to be shut down for further analysis.

A production Honeypot [11] is usually deployed in a business network, for the purpose of mitigating security risks in an organization. Production Honeypots protect a network in one of the following three manners. First, the production Honeypot does not only detect threats from the Internet, but if deployed on an internal network, may also detect insider threats; because insider criminals will access the Honeypot to obtain sensitive information. Second, the Honeypot can be a good barometer of threats against the network. Third, by making the Honeypot seem very valuable to an attacker, a good administrator can lure in a hacker, and have it spend its resources exploiting the Honeypot vulnerabilities, when they would otherwise be compromising the organization’s operational network.

Additionally, there are two types of Honeypots, *high-interaction* and *low-interaction*, depending on the level of access that they provide to the attackers. Table 1 summarizes the comparison between high-interaction and low-interaction Honeypots. High-interaction Honeypots provide real systems, applications, and services for attackers to interact with. The advantages of high-interaction Honeypots are that we can capture extensive amounts of information by giving attackers real systems to interact with. It enables us to learn the full extent of their behavior, everything from new root-kits to geographically-dispersed IRC sessions. Honeywall, Sebek, and CaptureHPC are some of the examples of high-interaction Honeypots [9]. On the other hand, low-interaction Honeypots provide emulated services and they are easy to install and deploy. These types of Honeypots capture limited information about the hackers and they are generally useful to understand a hacker’s specific activity. Dionaea, Honeyd, Nepenthes, and Google Hack are some of the examples of low-interaction Honeypots. Figure 1 shows a sample network layout with different types of Honeypot implementations.

Table 1: Comparison of high-interaction and low-interaction Honeypots

Low-interaction Honeypots	High-interaction Honeypots
Emulates operating systems and services	No emulation, real operating systems and services are provided
Easy to install and deploy. Usually requires simply installing and configuring software on a computer.	Can capture far more information, including new tools, communications, or attacker keystrokes.
Minimal risk, as the emulated services control what attackers can and cannot do.	Can be complex to install or deploy (commercial versions tend to be much simpler).
Captures limited amounts of information, mainly transactional data and some limited interaction.	Increased risk, as attackers are provided real operating systems to interact with.

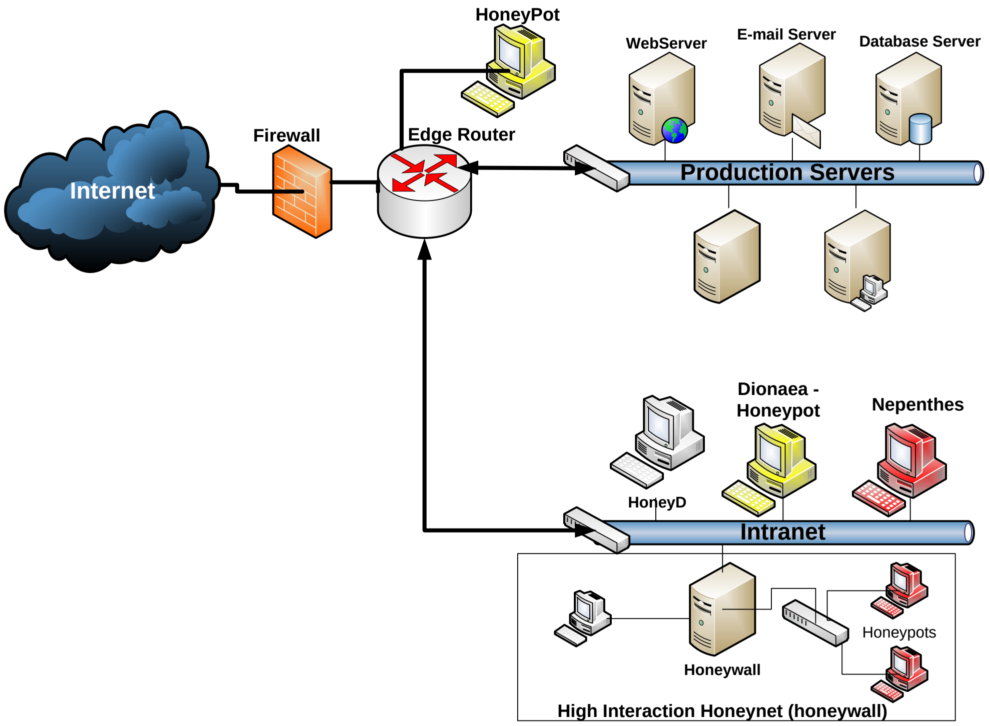


Figure 1: Various types of Honeypot implementations

A Honeynet gathers a large amount of network data encumbering its analysis. Various types of data are collected based on which Honeynet tool is used, e.g., Honeywall, Nepenthes, HoneyD, Dionaea, etc., and each tool uses its own format for data representation and storage. For instance, the Honeywall is a high-interaction Honeynet which has a built-in firewall, intrusion detection system (Snort), and Hflow daemon. The Honeywall acts as a layer 2 bridged gateway and is designed using a condensed Linux distribution [12]. It also has a kernel-level module which collects keystrokes and other activities in the Honeypot. Apart from these, the Honeywall also captures the packets and stores them using the PCAP format. The Honeywall runs a daemon known as Hflow which collects data from different sources and stores them in a MySql database. The information collected in the database includes the following:

- 5-tuples (Source and destination addresses, source and destination ports, protocol),
- Snort IDS responses – gives the relative threat level and also generates alerts,
- Passive OS fingerprinting – identifies the attackers' OS,
- Total bytes transferred, and
- Sebek data – data sent by the sebek client which captures the host activity

Table 2 presents a summary of the high-interaction and low-interaction tools surveyed as part of the literature review phase of this project.

Table 2: Summary of the high-interaction and low-interaction tools

Tool	Data Control	Data Capture	Data Analysis	Data Collection
High-Interaction Honeypots				
HoneyWall CDROM	✓	✓	✓	
Sebek		✓		
HIHAT			✓	
HoneyBow				✓
Capture-HPC		✓		
Low-Interaction Honeypots				
Nepenthes				✓
Honeyd		✓		
Honeytrap			✓	
HoneyC		✓		
Honeysnap		✓	✓	
Capture BAT			✓	
Honeymole				✓
Google Hack Honeypot		✓	✓	✓
Honeystick		✓		
Tracker			✓	

These deployments combine Honeynets with a virtual distribution technique, to significantly reduce the cost and time of deployment. Honeypot farms have a central system responsible for data capture and control. Figure 2 shows a Honeypot farm architecture. At each remote network site, a router or firewall redirects traffic destined to the Honeypot back to the Honeynet farm. The traffic is sent to and from the Honeypot farm over a VPN tunnel [11].

Honeypot farms can also be used to protect production hosts. This can be done by selectively redirecting the traffic to a Honeypot farm from the firewall rather than dropping it, since they do not physically run on the production hosts. This type of redirection is often called hot-zoning or bait-n-switch. Hot-zoning allows slowing down of an attack by deflecting the traffic to a Honeypot, and at the same time observing the attacker's techniques safely [11]. Deploying Honeypot farms has the following advantages [11]:

1. Rapid and short deployment time on new networks. Only router or firewall configuration changes are required to support the tunnel and routing changes.
2. Low level of involvement of the networks, since they do not need to configure or monitor Honeypots.
3. High level of control on the centralized site.
4. Hot-zoning support for protection of production hosts.

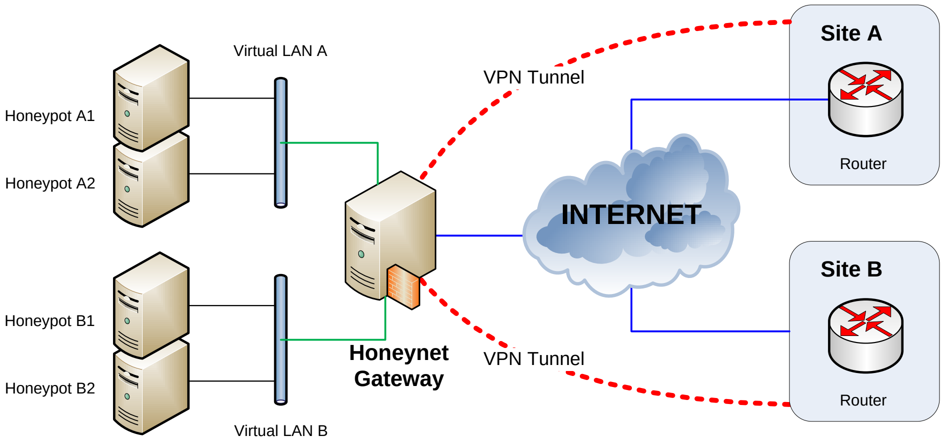


Figure 2: Honeypot farm architecture

On the other hand, it has the following disadvantages **Error! Reference source not found.**:

1. Virtual distribution can cause anomalies in latency that can be detected by an astute hacker.
2. Honeypot farms use routing rather than bridging, which causes much of the configuration complexity attempting to hide the honeywall from view.
3. They are complex to configure and require a good knowledge of networking.

The Honeynet Project [13] was founded in 1999 in order to define and provide resources to create Honeynets. The concept of trapping attackers was not very new but the project conceptualized the terminology and technical know-how, used by Honeynet technologies. A large number of people have since participated in the Honeynet Project. The first Honeynets were very simple, operating with a single computer to emulate a vulnerable system. Currently, the Honeynet project, including their regional chapters, and many corporate enterprises and universities, are implementing and developing Honeynets at several of their sites.

The Honeynet Project is based on an international chapter model. Chapters are smaller, self-governing entities that help manage the global structure of the organization. Each chapter is made up of full-members and contributors. Full-members are trusted members of the Honeynet Project that have the authority to vote on all Honeynet Project related issues. Contributors are members of the public that work with their local Chapter, but are not official members of the Honeynet Project.

3.2. Anomaly Detection

Anomaly Detection refers to a technique of detecting patterns that are different from the normal network profile. Such an approach helps to identify new or unknown patterns in any data set. The abnormal patterns within any data set are referred to as anomalies, outliers, exceptions, peculiarities, etc. [14]. Figure 3 shows the regions which are labeled as normal or outliers.

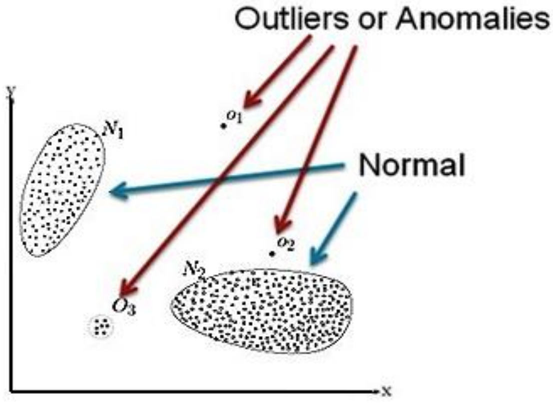


Figure 3: Anomalies or outliers [14]

Anomaly detection is a very useful concept due to its wide application in various fields. An anomalous behavior in the network could indicate a compromised machine or a machine transmitting sensitive data out of the network. There are various challenges in an anomaly detection approach such as defining the normal behavior and abnormal behavior, capturing most of the normal behavior, etc. Due to this reason, most of the existing anomaly detection schemes tackle only a specific problem [14].

There exist in the literature two main categories of anomaly detection for network traffic, i.e., volume-based detection techniques and feature-based detection techniques.

- **Volume-based detection techniques** [15-18]: A volume-based detection scheme is useful when identifying anomalies that cause large change of traffic volume, e.g., in a flooding attack or certain types of DoS attacks. The anomalies that do not cause large traffic volume changes cannot be detected by volume-based detection techniques.

- **Feature-based detection techniques** [19, 20]: The feature-based detection scheme uses the distributional changes of packet header details like IP addresses and port numbers to detect anomalies. Feature-based detection techniques require header inspection of each packet and this is time consuming and not applicable with real time constraints. Entropy has been used along with feature-based detection techniques in previous research work [17, 19-21].

In information theory, Entropy is defined as a measure of uncertainty or randomness associated with a random variable [22]. Entropy provides the measure of deviation in data items. Entropy can be used to detect anomalies in a given data set by finding out the variations in the entropy values. The entropy values of a sample of size *n* lies in the range$[0\ ,\log n]$. The entropy takes the minimum value of 0 when there is no variation in the data items, e.g., single IP address or port number; and it takes the maximum value of $\log n$ when all the data items are distinct or the variation is large. In entropy-based detection techniques, the entropy of a random variable X with possible values $\{x_1, x_2, x_3 \dots\dots x_n\}$ can be calculated as follows:

$$H(X) = -\sum_{i=1}^{n} P(x_i)\log P(x_i)$$

Suppose we randomly observe X for a fixed time window w, then $P(x_i) = m_i/m$, where m_i is the frequency or number of times we observe X taking the value x_i. Therefore:

$$m = \sum_{i=1}^{n} m_i$$

$$H(X) = -\sum_{i=1}^{n}(m_i/m)\log(m_i/m)$$

Where:

$H(X)$ = Entropy

m_i = number of packets with x_i as the traffic feature

m = total number of packets

The probability of occurrence of a traffic feature value in the observed traffic is computed as follows:

$$\mathrm{P}(x_i) = \frac{\text{Number of packets with } x_i \text{ as traffic feature}}{\text{Total number of packets}}$$

Here, the total number of packets *m* is the number of packets seen within a time window of fixed size *T*. More details about how the time window *T* is defined and what value is used for it will be provided later in this report.

3.3. Anomaly Detection Techniques for Honeynet Traffic Analysis

The various Honeypot implementations result in the collection of a huge amount of data of different types such as: packet-captures, tcpdump data, malicious binaries, keystroke logs, and URLs of malicious websites [23]. The raw data collected from a Honeynet can be used to provide further insights into the hacker’s activities. However, it becomes difficult to analyze the captured data without the use of automated analysis tools. The “needlestack” data overload, i.e., too much data and different types of data, is one of the main challenges for Honeynet analysts [24]. Honeynets are now used widely by many researchers and network operators to understand the vulnerabilities in the network. However, Honeypots collect a large amount of data from various data sources making it difficult to manage Honeypots and to analyze the collected data [25].

In addition, it is postulated that the true behavior of a hacker in action can only be realized if multiple Honeynets are deployed for data collection and analysis at various points of a network, whereupon the data can be correlated to construe the meaning out of an attacker’s behavior. One of the main objectives of this project is to study the effect of an anomaly classification technique for identifying malicious activities in Honeynet traffic. Such a technique shall help network administrators better utilize the Honeynet to understand different types of vulnerabilities, for the purpose of taking the necessary actions to protect their networks from malicious activities.

The success of a Honeynet mainly depends on the way the data is collected and analyzed to better understand the vulnerabilities in the network. In network security, anomaly detection plays a major role in detecting network security breaches or intrusions. Unlike its counterpart known as misuse-based or signature-based detection, the anomaly detection techniques are very useful in detecting new and unknown attack patterns. It is especially useful for detecting attacks such as the following [26]:

- New buffer overflow attacks carrying shellcode,
- New exploits,
- Intentionally stealthy attacks, e.g., using ADMutate to transform a shellcode, and
- Variants of existing attacks in new environments, e.g., worms using different file names as they propagate

Little work has been previously done to address the need for identifying malicious activities in the diverse data sets generated from Honeypots. The few approaches that exist mostly focus on detecting botnets and worm or virus outbreaks as they analyze traffic collected from low-interaction Honeypot sensors setup across the globe. Honeynet traffic is different from other types of network traffic as every packet that enters or leaves the Honeynet is considered malicious. Nonetheless, analyzing Honeynet data to identify malicious events is a challenging task and consumes a lot of time. Traffic collected by a Honeynet includes attack traffic, broadcast traffic, probes, and traffic from other local machines. The diversity in the traffic collected by a Honeynet, and the real nature of all such traffic (note that an attacker is unaware of the presence of a Honeynet) implies that novelty in analysis of such data is essential to achieve high rates of detection with low false alarms.

Lakhina et al. [19] proposed an anomaly detection method using traffic feature distributions in which they argue that distributions of packet features like IP addresses and ports are useful in detecting a wide range of anomalies in the network traffic. The authors stated that by using entropy along with traffic feature distribution, they can sensitively detect a wide range of anomalies; and it also helps in clustering the anomalies into different groups. In their experiment, they used network wide traffic as the data source as it contains various types of normal and anomalous traffic. The authors noted that identifying the nature of anomalies in a huge data set is a challenging task as the anomalies are a moving target. An anomaly detection system that depends on a predefined set of anomalies is inefficient as the anomalies are varying constantly. They pointed out that most of the anomalies affect the distributional aspects of traffic features like IP addresses and port numbers. The main difference between the method used by Lakhina et al. [19] and previous work is that they used distributions of traffic features, such as IP address and ports, to detect anomalies as opposed to using traffic volume. They noted that not all anomalies cause volume changes in traffic but most of them can be effectively detected using traffic feature distribution. The traffic features used by the authors are: source and destination IP addresses, source port, and destination port. The authors used the Principal Component Analysis (PCA) for traffic anomaly detection, which is used to separate the normal and anomalous behavior through dimensionality reduction. In our work, we are using traffic destined only to a Honeynet and we are using both traffic feature distributions and volume parameters to detect anomalies and classify malicious activities.

Nychis et al. [20] presented an interesting work by conducting an empirical evaluation of using Entropy for anomaly detection. The authors mainly focused on analyzing the effectiveness of using different traffic features and behavioral features distributions for anomaly detection. The behavioral features include the degree of distribution measuring the number of distinct source and destination IP addresses that each host communicates with. They conducted various experiments and showed that the IP address and port distributions are strongly correlated and provide similar detection capabilities. The behavioral and flow size distributions are less correlated and hence detect anomalies that are usually not detected by IP address and port distributions. The authors calculated the correlation between different feature pairs based on the entropy values to find the correlated feature pairs. They suggested that the selection of traffic feature distributions must be made carefully and it must not be restricted to port/address features. In our work, we are using the feature pairs that have the best detection capabilities for Honeynet traffic. The traffic features were compared and the best ones were chosen using the test data sets to classify the behavior of different types of malicious activities.

Kind et al. [27] proposed a new approach to the feature-based anomaly detection of Lakhina et al. [19]. In their proposed approach, the authors created histograms of the different traffic feature distributions and then modeled histogram patterns which are used to detect anomalies. They detect anomalies in four stages: select features and construct histograms, map into metric space, cluster and extract models, and finally classify the anomalies. In their approach, the authors use various traffic features like source and destination addresses, port numbers, TCP flags, etc. In this approach, PCA has been used for dimensionality reduction instead of differentiating between normal and abnormal traffic as done by Lakhina et al. [19]. The main difference of this approach is in the use of histograms to detect anomalies instead of using entropy. In our proposed work, we are using entropy values of different features along with the *k*-means clustering technique to identify anomalies in Honeynet traffic compared to using histogram patterns for clustering.

Ping and Abe [17] proposed an IP packet size entropy-based DoS detection scheme in which changes in the IP packet size entropy (IPSE) are used to detect possible DoS attacks. The authors note that various applications have different packet size profiles and this distribution changes in the presence of potential DoS attacks. The authors illustrated that various applications have default packet sizes with respect to request/response data. This is due to the fact that various services have default packet sizes based on the service provided. For example, FTP applications have 40 byte acknowledgements and a full packet data of 1500 bytes. In the presence of attacks, the generated packets are of identical sizes irrespective of the response from the victim. The threshold of entropy is obtained by self-learning from legitimate traffic data. After setting the threshold value, the entropy that exceeds this value indicates the presence of attack traffic. The IPSE approach was able to detect short term as well as long term attacks; which is an improvement over the traditional volume-based schemes. In our approach, we utilized the detection capabilities of volume-based schemes along with the feature-based detection schemes to identify the anomalous behavior.

Thonnard and Dacier [28] proposed a clustering-based approach to detect attack patterns in Honeynet data. In their approach, they specifically use time signature to cluster the Honeynet data. Time series is defined as a sequence of data points measured at successive times separated by uniform time intervals. They conducted experiments on large data sets collected from 44 worldwide distributed Honeypots. The attack source is identified as an IP address that targets the Honeypot on a given day with a certain port sequence. The network characteristics used by the authors include: (i) the number of virtual machines targeted on a platform, (ii) the number of packets sent to each virtual machine, (iii) the total number of packets sent to the platform, (iv) the duration of the attack session, (v) the average inter-arrival time between packets, and (vi) the associated port sequence. In our work, we are applying an Entropy based-anomaly detection technique to classify malicious activities in Honeynet data as opposed to using time signatures for clustering Honeynet data.

Al-Haidari et al. [21] proposed an entropy-based countermeasure against DoS attacks on firewalls. In their work, they used packet size entropy and the corresponding threshold values to distinguish between normal traffic and attack traffic. They have also illustrated that entropy-based scheme enhances the performance of the firewalls in terms of throughput, delay, and availability by isolating the attack traffic from the legitimate traffic.

Most other anomaly detection approaches which are used on production network traffic are not well suited for this type of traffic [28], since the Honeynet traffic is different from production network traffic. As stated before, a Honeynet is used by various organizations to proactively improve their security, and to detect malicious activity with relative ease. Another important use of Honeynets is to identify the tools, tactics, or behavior of different attacks, and for information dissemination amongst security agencies. In order to address these issues, we postulate that the use of an anomaly detection technique based on entropy and volume thresholds can be very effective in analyzing Honeynet traffic for studying the behavior of the attacker activity.

In our proposed approach, we use both entropy-based and volume-based detection schemes to identify anomalies in Honeynet traffic. Most of the other research work in the literature is focused mainly on comparing the effectiveness of these two techniques or to propose a technique based on either one of them. In our work, the main focus is to identify the best features of network traffic, for attaining a higher degree of accuracy in the detection process, whilst analyzing Honeynet traffic behavior.

4.0 RESEARCH METHODOLOGY

4.1. Approach for the Honeynet Design and Deployment

The team has taken an approach that combines the theoretical, developmental, and experimental aspects. We have studied the theoretical background of many issues related to security and Honeynets. Then, we have designed and implemented Honeynets based on the findings of this study. Afterwards, we have experimented with the deployed Honeynets to study, analyze, and improve them. In addition, the approach was incremental in that we have implemented the first Honeynet prototype in a lab environment, tested it, and learned from it. The second phase included a Honeynet implementation at the level of KFUPM. The project also led to national and international collaborations in the field of security in general and Honeynets in particular. The lessons learned were used to recommend a strategy and design for a Saudi wide Honeynet Project.

The team first started with an extensive survey about the state of the art in Honeynets designs including levels of interaction, data control, data collection, and data analysis. Section 3 of this report covers this in more details. Then, an inventory and comparison of open source tools for data control, collection, and analysis in Honeynets settings was conducted. The team also reviewed representative samples of exiting Honeynet projects around the world, in addition to the different methods and best practices to design and implement a Honeynet. The team has also set a mechanism for collaboration with the global Honeynet community researchers who are involved in investigating security within IT systems around the globe. This includes collaboration with CERT Saudi Arabia (CERT-SA) and with other Honeynet Project chapters.

For the purpose of learning from the experience of other Honeynet projects around the world, two members of our team visited the Malaysian Honeynet Project and the Malaysia Computer Emergency Response Team (MyCERT) which is part of Cybersecurity Malaysia. One objective of the visit was to learn more about the best practices used in the deployment of Honeynets. This has helped us take some major decisions in the course of the project. For instance, during the visit many issues were raised and discussed with regards to the Saudi Honeynet setup and configuration, and can be found in Appendix 1. The proposed recommendations were all taken in consideration by our team to improve the Honeynet we have deployed. One major decision that was taken is to move away from high-interaction Honeypots and focus more on the use of low-interaction Honeypots and more specifically Dionaea for the deployment of Honeynets.

In addition, we have initiated communication with CERT Saudi Arabia (CERT-SA), which has lead later to a visit by our team to their premises and the start of more collaboration with them, including a weekly report that is being sent by our team to CERT-SA about the attackers activities targeting KFUPM from outside. The report of the visit can be found in Appendix 2. CERT-SA has also recently become a chapter of the Honeynet Project. We have also initiated collaboration between CERT-SA, ITC at KFUPM, and the Saudi Honeynet (SAHNET) Project. A conference call was held between the three parties to discuss ways of collaboration including exchanging reports, agreeing on reports formatting, organizing security workshops, involving students, and other activities that support enhancing the security at the Kingdom level. Furthermore, the periodic reports sent by SAHNET to CERT-SA can be used for the purpose of presenting the analysis results in visual and textual format and for updating their web site reporting of online statistics on current attacks, worms, viruses, etc. (see http://Honeynet.org.sa/)

As part of the activities conducted by this project, two of our team members including the PI attended the Honeynet Project annual workshop. The detailed report of the workshop can be found in Appendix 3. The purpose of attending the workshop was to meet with members of various national Computer Emergency Response Teams (CERTs), experts from leading technology companies, and professors from various universities. The main objective of attending this event was to learn and discuss security issues related to our NSTIP funded project. During the workshop, we have established connections with many experts which have continued after the workshop. For instance, we are in constant contact with many members of the Honeynet project community to discuss different issues related to the implementation of our Honeynet project. This included exchanging traces with some other Honeynet chapters, which has happened after our return from the workshop. And, we have obtained a large set of traces from few Honeynet Project members worldwide.

In addition, and during the annual workshop, we got support from the developers of some of the main tools developed and being used by the Honeynet Project community, including Dionaea and Glastopf, which have been recognized by the Honeynet Project community as two of the most important tools to be used in this area. Dionaea is also considered by the community as the preferred tool. For instance, we had the opportunity to talk to the developers of these tools and get some questions answered and issues resolved. During this event, we also had the opportunity to present and discuss our research work related to the Honeynet traffic analysis and get feedback on it. In addition, we benefited from many presentations on security related issues, which have also helped us focus more on what is important with relation to Honeynets. Some of this information has been also shared with CERT-SA. Hands-on workshops allowed us to get a practical experience with reverse engineering methods used to analyze malwares. After the workshop, we have also been invited to become responsible for the translation to Arabic of the SANS newsletter, namely OUCH! which is widely read around the world. We have since then been responsible for the translation to Arabic of all the issues of this newsletter starting from May 2011.

One of the outcomes of this annual workshop was that we have invited two Malaysian CyberSecurity experts to visit KFUPM, share their experience with us, and provide consultancy with respect to our project. One of the visitors is a consultant for this project. The program of their visit included delivering seminars on topics related to Android malware, malware evolution, and PDF attacks. In addition, two full days training was delivered on web security including hands on sessions and a mini cyber drill. A one day workshop, attended by invitation only, on analyzing malicious PDF was also held, which included a walk through on how to analyze in-the-wild malicious PDF files. In addition, the program included a review and discussion of the SAHNET project and a SAHNET lab visit, meeting with KFUPM ITC security team, and meeting with KFUPM Faculty members and students. The detailed program of the CyberSecurity Malaysia Experts Visit can be found in Appendix 4. The feedback of these two experts allowed us to revise our deployment and the way we are using the data collected including what should be exchanged with the CERT-SA team.

Later, two members of our team also visited the Taiwan Honeynet Chapter. The purpose of the visit was to learn from the Taiwan Honeynet Project Chapter experience on deploying Academic Honeynets. This is due to the fact that they have built many Honeynets in the Taiwan academic network for the detection and collection of malware samples. In addition, the team learned about their designed platform, namely TaiWan Malware Analysis Net (TWMAN) to analyze malware samples. A detailed report about this visit can be found in Appendix 5.

Consequently, the team investigated techniques that can be used for the deployment and placement design and topology based on the best practices of other Honeynet Project chapters. This allowed us to propose a suitable approach for the design of Honeynets, including the network topology, hardware and software components, tools, and operating systems. We had also designed and deployed earlier in the project a first Honeynet prototype in a lab environment, and then tested it to see if it is able to collect, log, and capture famous worms, malware, viruses, penetration testing and scanning, and report alarms and statistics. We were able to detect the first worm in March 2010 by the Honeynet, which was a blaster worm. This phase also allowed us to study and address any issues with regards to the deployment of the Honeynet to ensure that the objectives of attracting attackers are met. In this phase, the team also studied the process of capturing, collecting, and storing of Honeynet data. Then, we deployed the 1st pilot run of the Honeynet in a real network with collaboration of ITC during the period of November-December 2010. The report of this pilot run can be found in Appendix 6.

Another major phase of the project was the design and deployment of the KFUPM Honeynet based on the lessons learned from the lab deployment. Based on what has been learned from the first pilot prototype, the team deployed Dionaea's LogXMPP with a centralized database which serves as a repository for collected data, logs, and statistics. Multiple Honeypots have been deployed at different locations in the KFUPM campus network for collection and analysis. More details about the KFUPM distributed Honeypot architecture using Dionaea's LogXMPP feature can be found in Section 5.1.3. The KFUPM Honeynet is used to capture attacks, analyze related information, and share any findings. Information surrounding security threats and vulnerabilities active in the KFUPM networks is therefore collected. At this stage, the team was also able to identify and report vital information and statistics on existing and new malware, viruses, or botnet attacks that are currently targeting the KFUPM networks, and possibly targeting other networks in the Kingdom. This has allowed us to perform analysis and fusion of the collected data. We have also deployed a test-bed for SURFids at KFUPM. SURFids is developed by SURFnet, which is a scientific and academic Internet service provider in the Netherlands. The idea behind SURFids is to place USB-based sensors in different LANs. Sensors in those LANs are connected to the Honeypot at a centralized location. The sensor gathers information about the detected illegitimate network activities. The connected parties can see the information about the detected malwares in their network via a web interface. More details about SURFids deployment at KFUPM can be found in Section 5.1.2.

Finally, the team investigated the extension of the KFUPM Honeynet to a larger and more complex Honeynet to be deployed in KSA, and proposed a strategy for the deployment of the KSA Honeynet. Then, a design of the Saudi Honeynet Project was recommended and that includes a proposal for the deployment of Honeypots kingdom wise at different locations in KSA networks. This design is based on the KFUPM distributed Honeypot architecture using Dionaea's LogXMPP feature that was discussed earlier and which can be found in Section 5.1.3. We also plan to communicate with other interested governmental and non-governmental partners to participate in deploying Honeypots at their locations.

Throughout the project, the team has documented all findings. In addition, progress reports were submitted every six months. The team also prepared papers for conference and journal publications reporting on the projects' findings, some of which have been submitted and others accepted and presented. Table 3 shows the approaches utilized for achieving all the project objectives.

Table 3: Approaches utilized for achieving the project objectives

Objective	Approach of achieving the objective
Objective 1:	Extensive survey of Honeynets designs, tools, and projects.
Objective 2:	Equipment Selection, Implementation, Experimental Verification, Testing of Operation.
Objective 3:	Pilot Project Feedback, Devise a KSA Honeynet Architecture, Design of Different Components, Provide a set of recommendations.
Objective 4:	Involvement of investigators in different project phases, Critical Review of Procedures.
Objective 5:	Acquire Equipment, Develop Experiments and Practical Examples.
Objective 6:	Gain Expertise, Provide Consultancy, and Initiate Collaborations.
Objective 7:	Establish a Center of Excellence in the Area of Honeynets.

Table 4 provides a mapping between the different phases of the project and the corresponding project objective(s) achieved.

Table 4: Mapping between project phases and project objectives

Phase	Achievements
Phase 1: Surveys and Literature Review	**Objective 1**
Phase 2: Honeynet Design	**Objectives 2 and 3**
Phase 3: Lab Deployment and Testing	**Objectives 2 and 4**
Phase 4: Data Analysis	**Objectives 2 and 4**
Phase 5: Honeynet Deployment on KFUPM Campus Network	**Objectives 2 and 5**
Phase 6: University, National, and International Collaboration	**Objectives 6 and 7**
Phase 7: Documentation	**Objectives 3 and 5**

In summary, the seven phases comprising our project and their interrelationship are depicted in Figure 4.

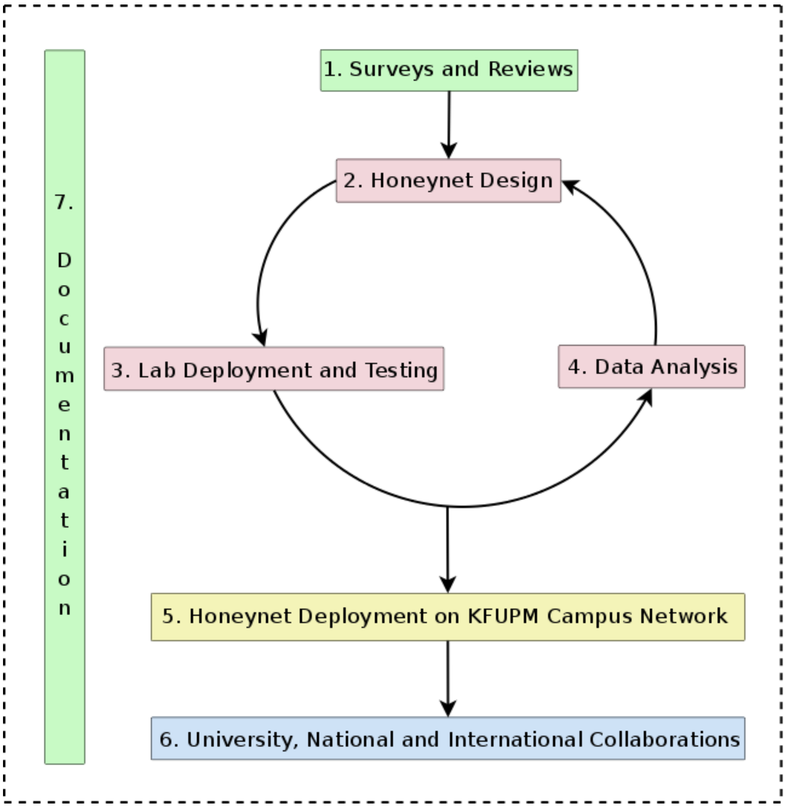

1. Surveys and Reviews
2. Honeynet Design
3. Lab Deployment and Testing
4. Data Analysis
5. Honeynet Deployment on KFUPM Campus Network
6. University, National and International Collaborations
7. Documentation

Figure 4: Project's phases and their interrelationship

4.2. Approach for Anomaly Detection in Honeynets

A Honeynet captures information that can be used by administrators to improve their network security, but the size of the data collected can be overwhelming [29]. Honeynets mainly depend on a signature-based detection scheme, manual analysis, and expertise to identify malicious activities. Honeynet traffic is different from any other network wide traffic as it has little or no production traffic. Any traffic that enters or leaves the Honeynet is suspicious by nature. However, in order to identify the different malicious activities in this traffic, manual analysis and expertise are needed.

As stated earlier, Honeynet traffic is different from other types of network traffic as every packet that enters or leaves the Honeynet is considered malicious. Based on this fact, we consider that anomalies that are classified as belonging to a given type are all malicious in nature. Nonetheless, analyzing Honeynet data to identify malicious events is a challenging task and consumes a lot of time. Traffic collected by a Honeynet includes attack traffic, broadcast traffic, probes, and traffic from other local machines which may not be always malicious, such as network discovery packets coming from windows based machines. The diversity in the traffic collected by a Honeynet, and the real nature of all such traffic (note that an attacker is unaware of the presence of a Honeynet) implies that novelty in analysis of such data is essential to achieve high rates of detection with low false alarms. Based on our Honeynets traffic analysis, we also found that significant changes in Honeynet traffic occurred only during malicious events, which essentially serves to identify anomalous activities within a given traffic profile.

There are very few anomaly detection techniques addressing the Honeynet systems' needs. Most of the Honeynet traffic is analyzed manually, which requires expertise to identify different types of attacks. The few existing approaches mostly focus on detecting botnets and worm or virus outbreaks as they analyze traffic collected from low-interaction Honeypot sensors setup across the world. Due to the fact that in a Honeynet most traffic that enters or leaves is considered malicious, other anomaly detection approaches applied to regular network wide traffic are not well suited for this type of traffic [28]. In order to address these issues, we proposed a simple and easy to use anomaly detection technique which can be used to identify malicious activities in Honeynet traffic and also to classify the behavior of various malicious activities. Thus, the project addressed specifically the classification part which focuses on mapping the different malicious activities to certain features' behavior using thresholds.

We proposed an anomaly detection technique which uses both feature-based and volume-based parameters to identify anomalies in the Honeynet traffic. The proposed approach uses a combination of packet header parameters entropies and volume changes to identify malicious activities. Our proposed method is composed of the following main steps:

1. Analyzing Honeynet traffic data and identifying the candidate features suitable for anomaly detection.
2. Selecting the features that provide good detection capabilities. These features will be taken from both those available in the literature as well as those obtained from a manual data analysis.
3. Devising and implementing a suitable anomaly detection technique.
4. Classifying malicious activities in Honeynet data based on the values (or ranges/thresholds) of the different features used by the proposed anomaly detection technique.

Figure 5 provides an overview of the proposed solution to classify malicious activities in Honeynet traffic. Using the proposed anomaly detection scheme in Honeynets will greatly improve the data forensics and the detection of unknown and new attacks. Although the focus of this part of the project was on the classification of malicious activities in Honeynets, the ultimate objective is to be able to identify similar malicious activities in any Honeynet traffic, including large data sets, which can then be filtered out to focus more on new types of attacks (or zero-day attacks).

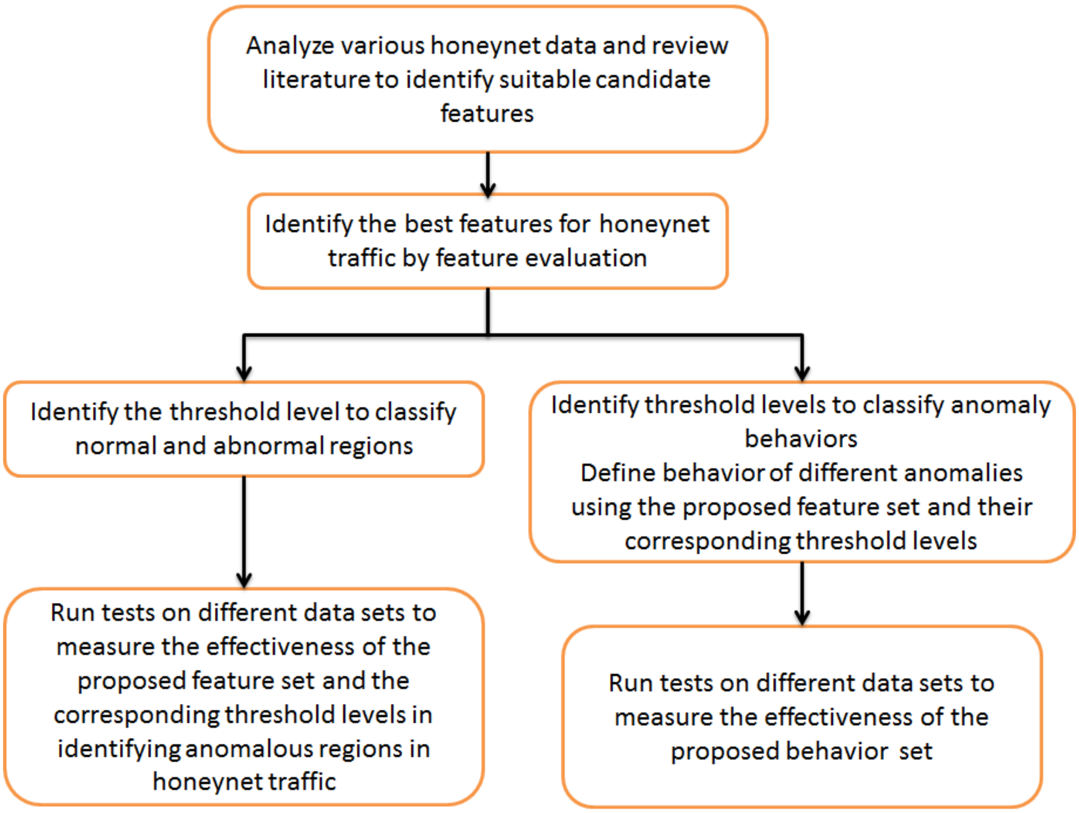

Analyze various honeynet data and review literature to identify suitable candidate features
Identify the best features for honeynet traffic by feature evaluation
Identify the threshold level to classify normal and abnormal regions
Identify threshold levels to classify anomaly behaviors
Define behavior of different anomalies using the proposed feature set and their corresponding threshold levels
Run tests on different data sets to measure the effectiveness of the proposed feature set and the corresponding threshold levels in identifying anomalous regions in honeynet traffic
Run tests on different data sets to measure the effectiveness of the proposed behavior set

Figure 5: Proposed solution for classifying malicious activities in Honeynet traffic

5.0 RESULTS AND DISCUSSION

The project has two major aspects. The first one is related to the Honeynet design and deployment, the lessons learned, and the architecture proposed. The second aspect is research-based and includes the contribution of our team in the area of applying different techniques for the analysis of Honeynet traffic. This section will summarize our findings in both aspects of the project.

5.1. Honeynet Design and Deployment

5.1.1. Design and Deployment of a Pilot Honeynet

The Saudi Honeynet project team undertook the 1st pilot run in a real network deployment on KFUPM premises during the months of November-December 2010. The Honeypot tool used as the standard platform to run the pilot experiment was *Dionaea* since it is widely used and is the recommended low interaction Honeynet by the Honeynet Project. The pilot deployment consisted of two phases. During the first phase (November 6th & 9th, 2010), we placed the Dionaea-based Honeypot on the public Internet within the Information Technology Center (ITC) premises at KFUPM, whereas in the second phase (November 17th, December 3rd, 2010), the Honeypot was placed on the ADSL network, which facilitates Internet services for the KFUPM faculty housing area. During this activity, we collected network traffic and results therein; some of which are discussed in the summary report found in Appendix 6.

As shown in Figure 6, the location of the Dionaea Honeypot was on the Internet end of the ITC network, for the first pilot run. In this scenario, we placed the Honeypot outside the KFUPM network, so that if any malicious activity takes place, the university network will not be affected by it. The motive behind having the Honeypot outside the KFUPM network was to receive Internet traffic directly (unfiltered and unaltered), rather than coming through a firewall and a NAT router. For scenario 2 as illustrated in Figure 7, we placed the Honeypot in the faculty housing connected to the Internet through the ADSL network. During our initial runs, the ADSL network was not placed behind the firewall and at that time we recorded activities of certain viruses that had been active within the network.

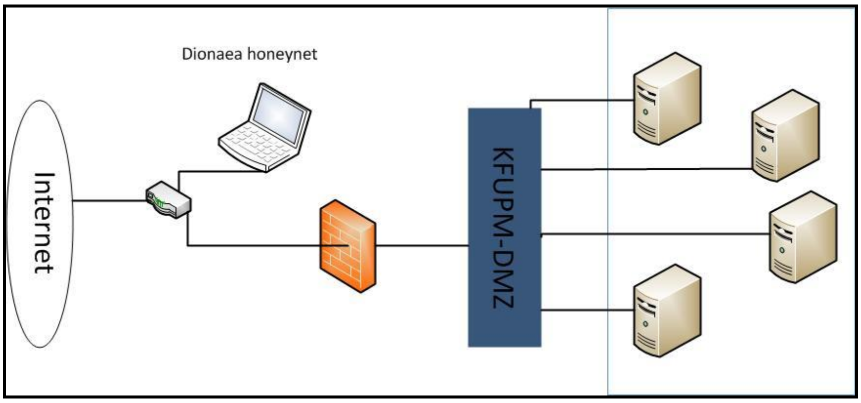


Figure 6: The Pilot deployment map on the KFUPM network

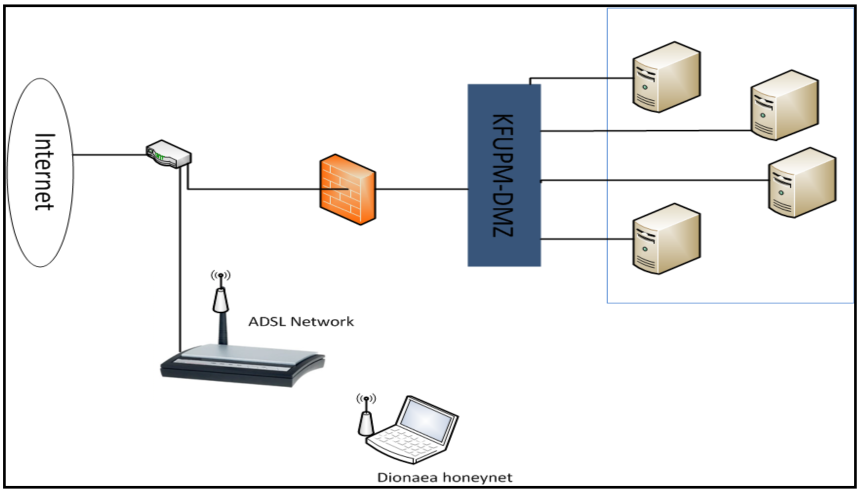


Figure 7: The ADSL deployment scenario for the pilot run

The types of activities seen on the Honeypot during this pilot test phase include:

1. SIP port scanning – IP: 216.55.161.16, 218.61.234.246, 221.231.150.67, 202.5.168.213
2. IP from china trying (attempting with many passwords) to log into the MS- SQL service offered by Dionaea. There were ~300 SQL login attempts made by the attacker source IP 220.168.169.100.
3. Port scan from ADSL network from IP 196.15.58.160.
4. Sunday (sundayddr) SIP scanning worm
5. Phpmyadmin attack
   Vulnerability Scanners

5.1.2. Design and Deployment of a Pilot SurfIDS Honeynet

SURFids is developed by SURFnet [30], which is a scientific and academic Internet service provider in the Netherlands. The idea behind SURFids is to place USB-based sensors in different LANs. Sensors in those LANs are connected to the Honeypot at a centralized location. The Honeypot gathers information about the detected illegitimate network activities. Via a web interface the connected parties can see the information about the detected malwares in their network. SURFids has divided the whole design into three major components, as shown in Figure 8:

1. Centralized Server or Tunnel Server
2. Logging Server
3. Sensor

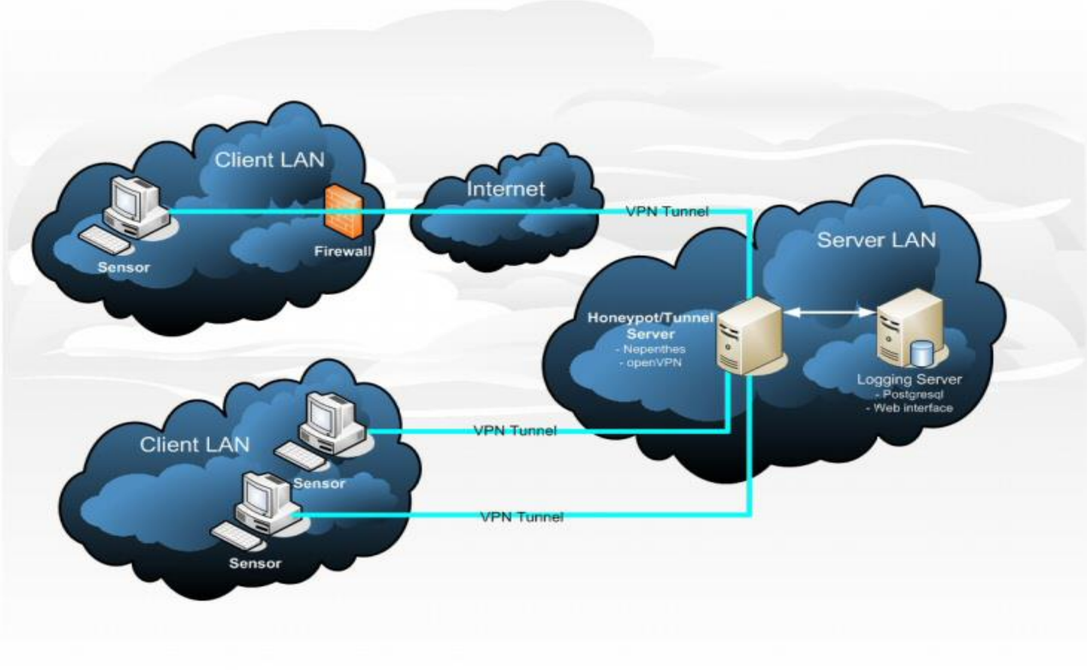


Figure 8: SurfIDS Components [30]

The first two components, i.e., the tunnel server and the logging server, are centralized components of our network. However, the sensor is part of a network which can be anywhere on the Internet or inside any cooperate network. A sensor only needs an Internet connection to connect to the tunnel server. The centralized server has two building blocks, namely the tunnel server and the Honeypot. The task of the tunnel server is to receive a VPN connection request from the sensors and create a VPN tunnel with the sensor. Once the tunnel is created, the tunnel server spawns a bridge device on the sensor's LAN and requests an IP from the DHCP server in the sensor's LAN. The other end of the bridge will be connected to the Honeypot. Now, the Honeypot is available in the sensor's LAN through an Ethernet bridge created over the VPN tunnel. So, in this way we will virtually place a Honeypot inside the sensor's LAN. Any spread of worms inside the sensor's LAN or any attacks on the Honeypot will be scanned and recorded by the Honeypot. Consequently, the Honeypot will add these threats to the centralized database.

The logging server provides the services of a database server. The logging server also presents these database records through a web interface. Therefore, the analysis of these threats becomes very easy. The reason for creating a separate log server is that if our Honeypot got compromised, then all the attack records will be safe on another machine. The web interface will provide an easy way to view and analyze the results. The web interface also ease a sensor's management by providing options such as start, stop, restart and ignore the sensor.

In SURFids, the sensor is actually a bridge device which performs two tasks. First, it connects to the centralized server, i.e., the tunnel server, through a VPN tunnel. Then, the sensor creates a bridge device in its own network. This bridge device will have its own IP address, and it connects the Honeypot inside the tunnel server to the client LAN. Any attacker inside the client LAN will consider this bridge device as a separate host and will find it more attractive to attack as this device is representing a Honeypot with very common vulnerabilities.

The tunnel server needs to be on the Internet with a static global IP address, as it needs to listen for incoming connections from sensors from anywhere on the Internet. Also, the tunnel server needs a local LAN connection, as the Honeypot running on the tunnel server needs access to the logging server to access the database and store network activities/attacks records. Since the logging server holds all the data, it needs to be protected and hence it is located inside a LAN. The sensor is very lightweight in terms of software installation, as its task is only to create a VPN tunnel and a bridge device.

For the SURFids deployment at KFUPM, we have used one server with debian Linux distribution system to hold the tunnel server and the logging server. The server was placed in a network security lab at KFUPM. For the tunnel server, we open ports 1194 and 4443 on the firewall to receive incoming connections. Port 1194 is used by the OpenVPN server to receive incoming connections. Port 4443 will be used by the apache2-SSL server. The SSL webserver normally uses port 443, but as we also have Dionaea running on the tunnel server, we cannot use port 443, and that is why we have used port 4443 instead.

The sensor has also been created on the same server. However, sensors can be placed anywhere inside the campus LAN. As sensors are virtual machines, they can run on any machine using the VMware player. To gather data from different locations, we can either connect multiple sensors with the tunnel sensor or use the same sensor at different locations. For our deployment at KFUPM, besides the sensor in the main server, we have installed a remote sensor in another lab within the campus, and it was connected to the tunnel server through a VPN tunnel. In addition, another remote sensor has been installed on a laptop. The laptop has the ability to be placed in different locations at KFUPM. The *Dionaea* Honeypot has succeeded in monitoring the activity on different ports.

5.1.3. Design and Deployment of a KFUPM Distributed Honeypot Architecture

Based on the feedback we got from the first pilot run and from the experts we met in the Honeynet Project workshop, we have designed and deployed a distributed Honeypot architecture within the KFUPM campus using Dionaea's LogXMPP module. Dionaea comes with different log submission modules, one of which is logxmpp. Using this module, Dionaea can post all the activities to an XMPP server [31, 32]. The Extensible Messaging and Presence Protocol (XMPP) is an open-standard communications protocol for message-oriented middleware based on Extensible Markup Language (XML).

Dionaea can post all the activities to two Multi User Channels (MUCs) on an XMPP server, *anon-events* and *anon-files* (anon stands for anonymous). Anon-events contain all the connection related information while anon-files contain all the malware samples offered to the Honeypot by the attackers. What makes this method unique is the XMPP server's IM capability, and many non-Honeypot or non-contributing users can also join these MUCs. This makes this technique a suitable candidate for the distributed architecture.

Figure 9 shows the currently running distributed architecture of the Honeypots at KFUPM. In this architecture, we are running prosody as the XMPP server. Prosody is a modern flexible communications server for Jabber/XMPP written in Lua and is licensed under the permissive MIT/X11 license. All the Honeypots create a secure connection with the XMPP server, which hosts the anon-events and anon-files channels. As soon as any Honeypot receives any new connection or malware sample, it posts this as a message to the respective channel. A backend script is running on both channels, and as new information comes from Honeypots, it reads the information and saves it in the postgresSQL database. PostgresSQL is a powerful, open source object-relational database system. A perl/django based web interface reads this data and posts the information on the web. This allows monitoring the activities across all the Honeypots in real time.

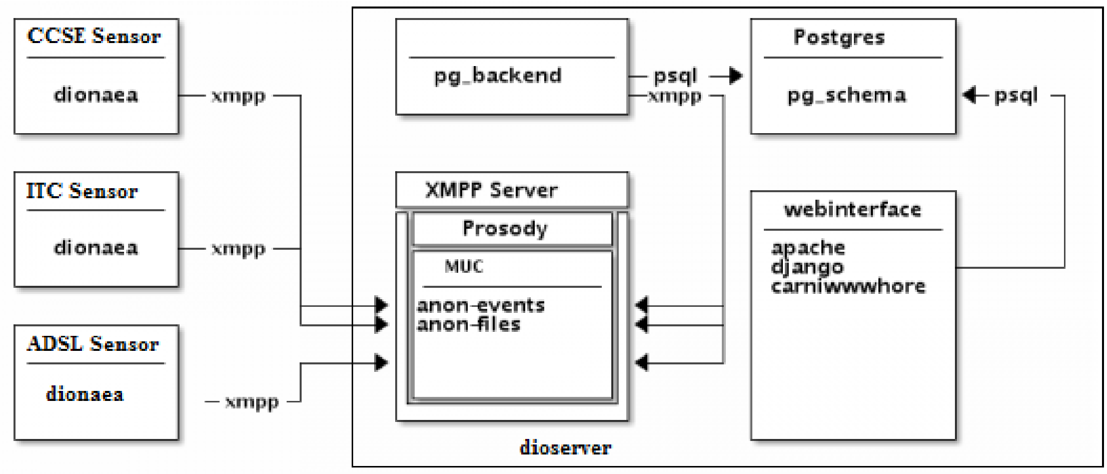


Figure 9: KFUPM distributed Honeypot architecture

Currently at KFUPM, we are running a small setup with three dionaea Honeypots. They are named after their location within KFUPM. The CCSE sensor is placed in the College of Computer Science and Engineering network, and this network has the maximum protection against attacks. The ITC sensor is placed in the Information Technology Center, but its network location is outside any firewall, so this sensor is exposed to the public and therefore receives the largest number of attacks. The ADSL sensor is placed inside the ADSL network provided to faculty housing. Although this network is also protected by a firewall, the security level is not as high as that of KFUPM enterprise network. Each Honeypot is a dionaea based with logxmpp feature enabled, while dioserver is our central server hosting XMPP server and the webserver. Each Honeypot submits all the events and files to the XMPP server on dioserver. Then, the backend script running on the dioserver reads all the messages from the Honeypots and adds them to the database. Since the database is connected to the web interface, any event posted to the XMPP server becomes available to the web interface immediately allowing real-time network monitoring.

5.1.4. Proposed Design for Honeynets at the Level of Saudi Arabia

The distributed architecture discussed in section 5.1.3 above is scalable and hence more Honeypots can be added without any modifications. Therefore, we believe that this makes such architecture suitable for deployment across the Kingdom of Saudi Arabia. Initially, we can place a single sensor in major cities or in major universities across the Kingdom of Saudi Arabia (KSA). Gradually, we can add public sector and private organizations to this architecture. One of the main features of this architecture is the real time monitoring web interface. As soon as any attack or malicious activity is reported by any of the Honeypots, it will be available on the web interface. As such, the task of security analysts becomes easier in protecting their organization and in raising the awareness of such attacks.

As a second option, a similar deployment of the SurfIDS Honeynet can also be extended to the KSA level. We need a main server to hold both the logging and the tunnel servers. The main server should have a fixed IP address to enable the sensors to connect to it. The main server should also be configured to collect data using the same setup used for the KFUPM deployment scenario. Then, one or multiple sensors should be installed in each city in KSA. The sensors should be configured to connect to the fixed IP address of the main server. When the sensors are connected to the main server, the main server will be able to monitor and gather traffic from all the sensors across the Kingdom.

5.2. Anomaly Detection in Honeynets

A Honeynet captures substantial amount of data and logs for analysis in order to identify malicious activities perpetrated by the hacker community. The analysis of this large amount of data is a challenging task. The main aim of the work presented in this section is to employ an anomaly detection technique to classify different types of malicious activities present in Honeynet. In particular, our technique utilizes both feature-based and volume-based schemes to classify Honeynet data and identify malicious activities in the Honeynet traffic. A detailed analysis of various traffic features is carried out and the most appropriate ones for Honeynet traffic are selected. The classification of malicious activities is achieved by applying entropy-based distributions and traffic volume distributions. Entropy-based distributions are used for feature-based parameters while traffic volume distributions are used for volume-based parameters. The behavior of various anomalies or malicious activities is classified using the selected features and their respective threshold values. Then, we propose a mapping between the various anomalies and their associated behavior, which can be further used to identify similar anomalies in other Honeynet data sets. Finally, we evaluate the effectiveness of the selected Honeynet traffic features in identifying the malicious activities in Honeynet traffic by using entropy distributions and volume distributions, along with their corresponding threshold levels. The proposed scheme proves to be effective in identifying most types of anomalies seen in Honeynet traffic.

5.2.1. Honeynet Test Data

In order to identify anomalies in Honeynets, we first need to analyze different Honeynet data sets to understand the difference between the normal and abnormal behaviors. Honeynet traces were collected mainly from the honyenet.org site which includes the scan of the month (SOM) challenges and Forensic Challenges released by the Honeynet Project organization [33]. The other source of traces is the hack.lu 2009 Information Security Visualization Contest [34]. The Honeynet traces that were used are listed in Table 5.

Table 5: Honeynet traffic test data sets used for analysis

Traffic Data Set Name & Source	Description	Traffic Details
Pcap Attack Trace, Honeynet.org – Forensic Challenge	The network traffic captured in the file attack-trace.pcap relates to an automated malware attack that exploits the Windows Local Security Authority (LSA) Remote Procedure Call (RPC) service.	348 packets Total duration: 16 sec
Scan 28 - Honeynet.org – Scan of the Month	This trace was collected by the Mexico Honeynet Team - Italian blackhats break into a Solaris server then enable IPv6 tunneling for communications.	Two traces:Day1: 18843 Packets – 24 Hours Day 3: 123123 Packets – 24 Hours
Scan 14 - Honeynet.org – Scan of the Month	This trace is about a successful Windows NT attack.	6707 packets Total Duration: 20 Hours
Scan 19 - Honeynet.org – Scan of the Month	This is a trace of Redhat Linux 6.2 Honeypot compromise.	24440 packets Total Duration: 23 Hours
SSH Based Honeypot trace - Information Security Visualization Contest - hack.lu 2009	This dataset was collected from an SSH based Honeypot. It includes anomalies such as network scans, rootkit file transfers, IRC traffic, etc.	4323191 packets Total Duration: 12 days

The traces provided by the Honeynet Project organization are instances of real compromises that were captured by different Honeynet Project chapters. The main reasons for releasing such challenges are to help the network security analysts to hone their forensic and analysis skills to get an in-depth knowledge of real attacks. These traces proved crucial in our work to characterize and identify the important features in the Honeynet traffic. As these traces are collected in a real environment and specifically in a Honeynet setup, it was of more importance to our work. These traces were analyzed to identify the suitable characteristics/features that can be used for anomaly detection. The analysis was done using tools such as Wireshark[35] and NetMiner[36].

The lists of features that were recorded from the literature and identified during test data analysis are stated in Table 6 and Table 7.

Table 6: List of feature-based parameters selected from test data analysis and literature

Traffic Feature-Based Parameters	Description
Source IP Address Entropy [19]	This parameter indicates the entropy of the unique IP addresses of incoming connections to the Honeypot.
Destination IP Address Entropy [19]	The destination IP entropy indicates the number of external connections initiated by the Honeypot.
Source Port Entropy [19]	This attribute indicates the number of source ports that are visible during each interval.
Destination Port Entropy [19]	This parameter indicates the number of destination ports visible during each interval.
Indegree [20]	Number of distinct Hosts that connect to the observed host. This parameter indicates the number of incoming connections to the Honeypot.
Outdegree [20]	Number of distinct IP address the observed host connects to. This feature measures the number of outgoing connections from the Honeypot.
Packet Size Entropy [17]	Various packet sizes visible in the network traffic.
Application Protocol Used	Application protocol seen during a conversation (e.g., SSH, SMTP, FTP, etc.).
Origin of IP address – Country	The distribution of countries from which the observed host gets connections.

Some of the features which provided redundant information were eliminated such as the application protocol since it is related to the port used. Similarly, instead of using the average packet sizes for different transport protocols, we choose the average payload size. A summary of the traffic features used for further analysis of the Honeynet traffic is presented in Table 8.

Table 7: List of volume features selected from test data analysis and literature

Volume Features	Description
Average number of bytes per TCP packet per minute [37]	Average TCP packet size per minute.
Average number of bytes per UDP packet per minute [37]	Average UDP packet size per minute.
Average number of bytes per ICMP packet per minute [37]	Average ICMP packet size per minute.
Sum of average packet size [37]	Aggregate sum of packet size average.
Total Payload Bytes	Total bytes seen in the five minute interval.
Average Inter-arrival times	Average inter-arrival time of packets in five minute interval.
Average Payload Size	Average packet size seen during the five minute interval.
Total Packets	Total packets seen during the five minute interval.

Table 8: Traffic features used for a detailed analysis

Traffic Features	Volume Features
• Source IP Address • Destination IP Address • Source Port • Destination Port • Packet Size Distribution • Indegree & Outdegree	• Average Packet Inter-arrival Time • Total Payload bytes received during the interval • Average Payload size during the interval • Average number of Packets received during the interval

5.2.2. Data Sets Analysis and Features' Evaluations

The real Honeynet traces obtained from Honeynet.org were used to test the effectiveness of each individual feature. The candidate features were evaluated based on the traffic distributions seen during the anomalous events. The features were also evaluated based on their ability to differentiate between normal and abnormal traffic. The entropy distributions were obtained by calculating the entropy values of each feature for every five minutes interval. Figure 10 shows the sliding window concept that was used to gather entropy values in overlapping intervals so that any valuable information is not missed in cases where an anomaly overlaps across multiple intervals.

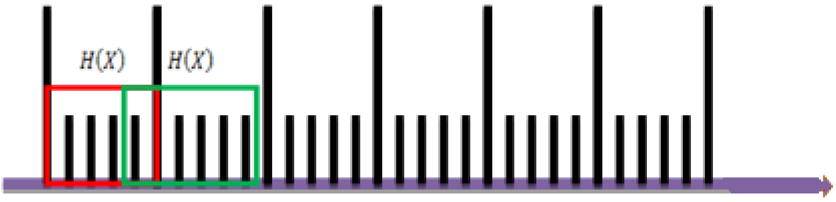


Figure 10: Sliding window used for calculating entropy

The entropy values of each feature were recorded and further manual analysis of the trace was performed to identify the normal behavior and anomalous behavior. Initially, all the features listed in Table 8 were tested, and later the best features that provide better detection capabilities were selected.

5.2.2.1. Data Set: Scan 28

This data set was published in the scan of the month challenges in the Honeynet.org website. The trace was collected by the Mexico Honeynet Team, and it is about Italian blackhats that broke into a Solaris server and then enabled IPv6 tunneling for communication. It is composed of two days of collected traffic, i.e., Day1 and Day3. The Day1 traffic is about the Honeypot being compromised and the Day3 traffic consists of the IPv6 tunneling enabled by the blackhats for communication.

Day1 Traffic

The destination port entropy of Day1 traffic does not show much activity in the first 9 hours after which there is a drastic change in the traffic behavior as shown in Figure 11. When we check the volume feature, i.e., the total packets in the interval after the 9th hour, it is clear that there was a malicious activity as shown in Figure 12. The manual analysis of the PCAP trace reveals that the Honeypot was probed for a specific vulnerability and then compromised during this time. A similar analysis was performed for other features to identify those that had better detection capabilities. The features that gave a clear indication of anomaly are destination port entropy, source port entropy, total payload bytes, and total packets. The packet size entropy also showed the change in behavior but it does not help in understanding the anomaly behavior.

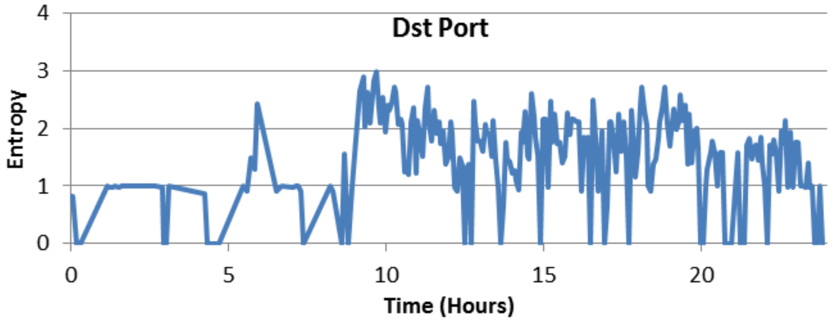


Figure 11: Destination Port Entropy in Day1 traffic of Scan28 data set

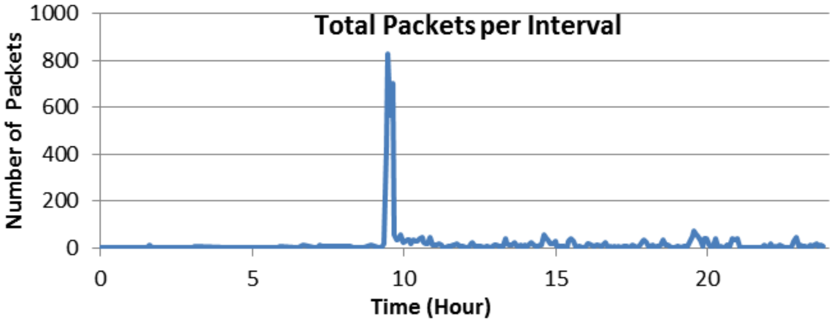


Figure 12: Total Packets per Interval in Day1 traffic of Scan28 data set

Day 3 Traffic

The Day3 traffic shows less activity in the initial hours, but around the 6th hour the traffic pattern changes. The manual analysis of the trace shows that the hacker had initiated an IRC connection to an external server. The source port entropy plotted in Figure 13 shows a drastic increase in the entropy value around the 15th hour. Also, a port scan activity was recorded which can be attributed to the peak in the source port entropy.

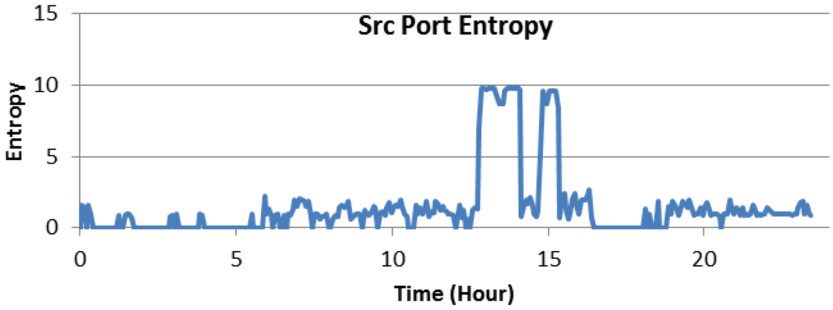


Figure 13: Source Port Entropy in Day3 traffic of Scan28 data set

The destination IP entropy and outdegree do not give a very clear picture of the changes in the traffic. The dominant features that were helpful in detecting the malicious events in this trace are the source port entropy, destination port entropy, total payload bytes, and total packets.

5.2.2.2. Data Set: Scan 14

This trace is about a successful Windows NT machine attack. The attacker exploited a vulnerability in Microsoft® Data Access Components (MDAC) that could allow a web site visitor to take unauthorized actions on a web site hosted using the Internet Information Server. The destination port entropy plotted in Figure 14 shows a different behavior during the period when the target machine was being compromised. The volume feature total payload bytes plotted in Figure 15 shows the intervals when large data or files were transferred to the target machine. Both total packets and total payload bytes show a large variation when some data transfer took place.

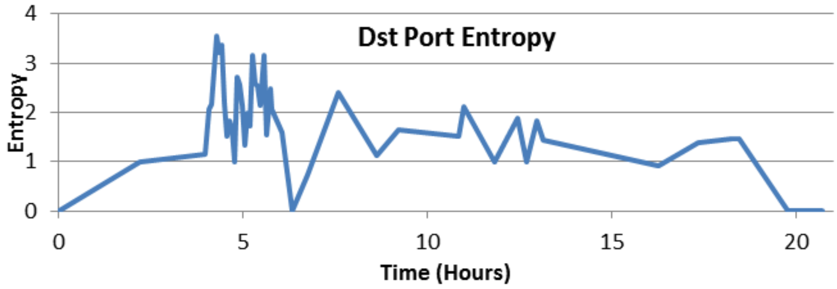


Figure 14: Destination Port Entropy for Scan14 challenge

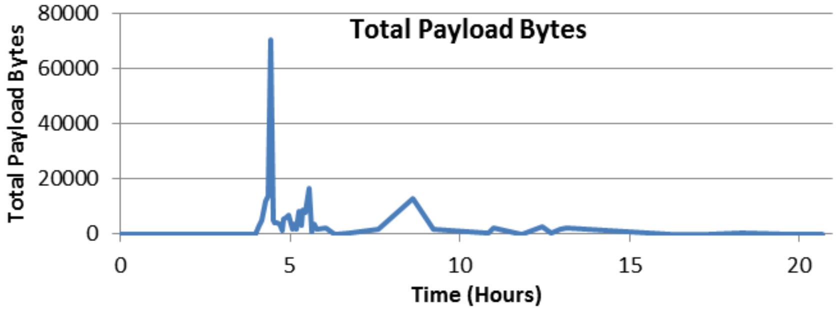


Figure 15: Total Payload Bytes for Scan14 challenge

It is clear from this trace that even in a short duration trace it is possible to detect the anomalies using the entropy of traffic features and the value of volume features.

5.2.2.3. Data Set: Scan 19

This trace was captured during a Red Hat Linux Honeypot compromise. The attacker exploited the vulnerability in the wu-ftpd (Washington University FTPD software) package.

After compromising the machine the attacker used three different modes to connect and execute the commands. The destination port entropy plotted in Figure 16 shows that there was not much traffic for nearly 20 hours and then there is a sudden dip in the entropy followed by a sharp increase. The dip in the entropy occurred when the attacker tried to exploit the specific vulnerability in the Honeypot. The importance of volume features is clear in this trace as they help in understanding the attacker's behavior during a system exploit. The other parameters like outdegree shown in Figure 17 and indegree are not very useful in giving a good understanding of the behavior.

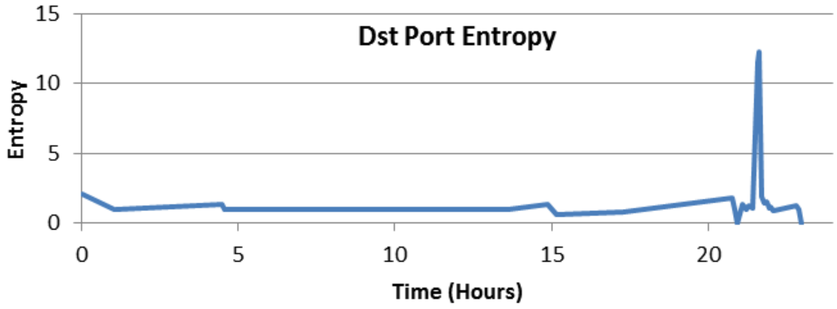


Figure 16: Destination Port Entropy for Scan19 challenge

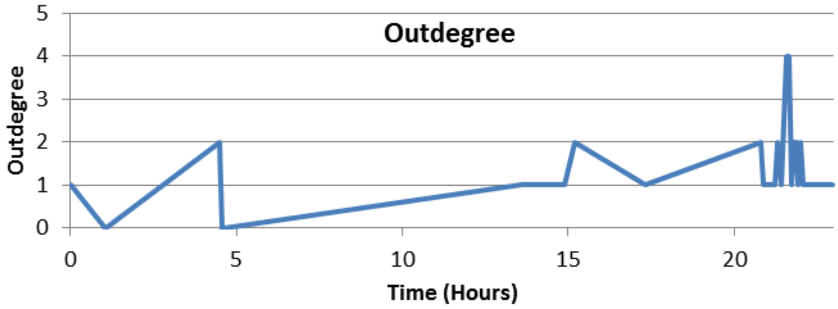


Figure 17: Outdegree Distribution in Scan 19 Trace

5.2.2.4. Data set: SSH based Honeypot Traffic

The feature analysis tests were also carried out on a large data set collected from an SSH based Honeypot which includes 12 days of traffic. The data set includes mainly SSH traffic and an unknown number of anomalies. The traffic includes anomalies such as network scans, rootkit file transfers, IRC traffic, etc. The destination IP entropy shown in Figure 18 indicates the number of external connections initiated by the Honeypot. The peaks indicate that the Honeypot initiated a large number of connections during that interval. The high value of Destination IP entropy indicates that the Honeypot was scanning the network.

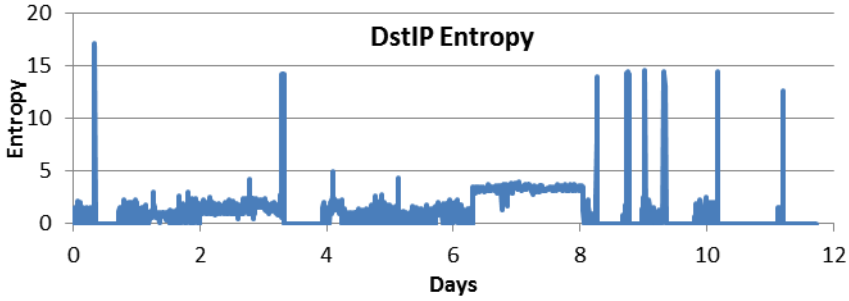


Figure 18: Destination IP entropy of SSH based Honeypot trace

The indegree shown in Figure 19 does not show all the anomalies and due to this fact, this feature was not selected for anomaly detection.

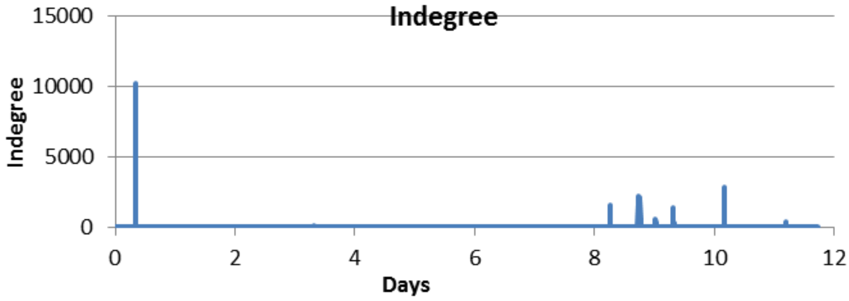


Figure 19: Indegree distribution of SSH based Honeypot trace

Volume based features like total payload bytes also helped in understanding the behavior and the anomalous events. Figure 20 shows that, before a network scan event begins, a large data transfer took place. When we manually analyzed the trace we found that this was related to a malicious file transfer which was later used to initiate the network scan activity.

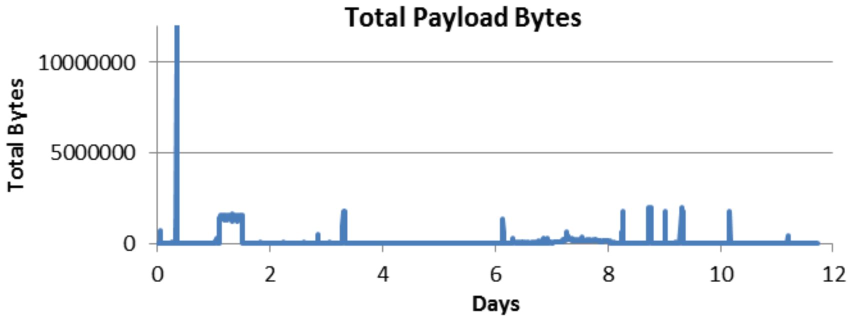


Figure 20: Total Payload Bytes distribution of SSH based Honeypot trace

5.2.2.5. Combining Pairs of Features to Detect Anomalies

Using individual features helps only in detecting certain anomalous events and it does not give a clear understanding of the anomaly that occurred. To get a better understanding of the behavior of the anomaly, we need to look into a combination of features. This is useful to detect certain anomalies that were not visible using a single feature. A number of combinations of the above listed features were tested to identify the useful features' combinations and get a better understanding of the anomalies.

The destination IP entropy and the destination port entropy show visible groups, i.e., clusters, indicating events with similar behaviors. In Figure 21, the group with high destination IP entropy and low destination port entropy indicates a network scan where a large number of IP addresses are being scanned for the same port. The cluster with high destination port entropy and low destination IP entropy is related to a port scan activity.

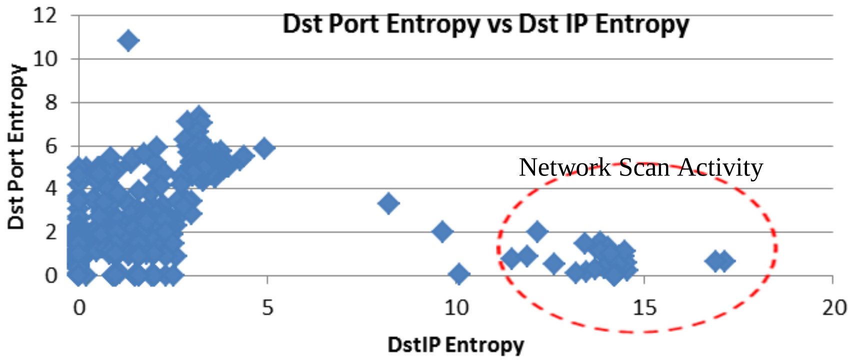


Figure 21: Destination Port entropy and Destination IP entropy combination of SSH Honeypot trace

The combination of source IP entropy and source port entropy plotted in Figure 22 shows that during a network scan, the source IP address entropy value is small due to the fact that only one IP was scanning the network.

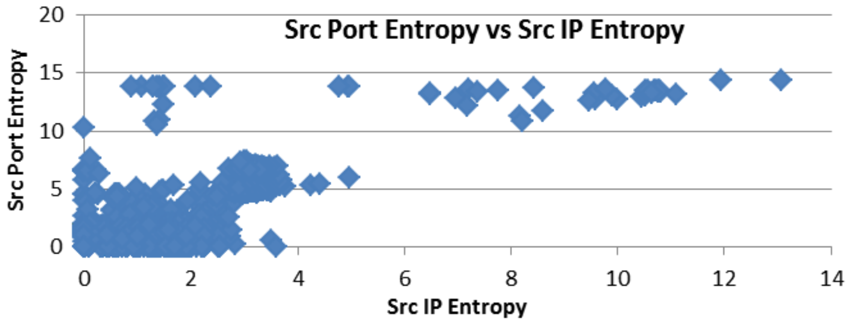


Figure 22: Source IP entropy and Source Port entropy combination of SSH Honeypot trace

5.2.2.6. Combining Three Features to Detect Visible Anomalous Groups

When combining different features, we can see different patterns that can help us detect anomalous regions as well as normal regions. Using three features helps in getting a better visualization of the different clusters present in the Honeynet data. We performed various tests using different combinations of the features to identify those features that provide the best distinction between a normal behavior and outliers by showing distinct clusters.

The combination of source IP, destination IP, and destination port is shown in Figure 23. This combination does not show many cluster regions because the source IP entropy and destination IP entropy have a similar behavior.

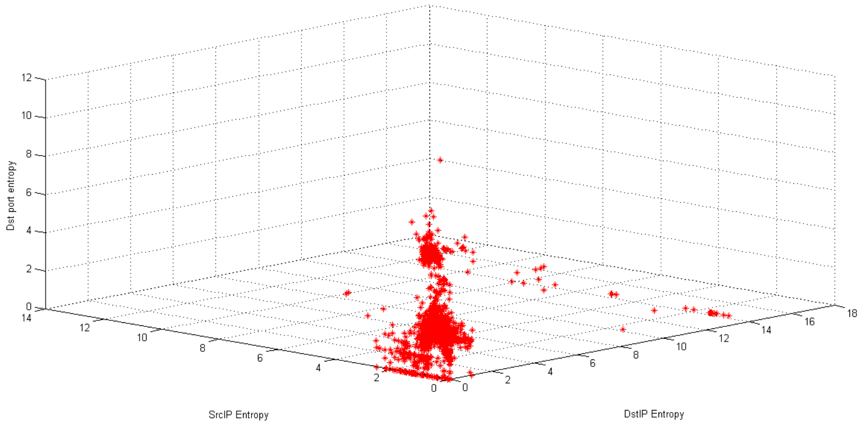


Figure 23: Combination of Destination port, Source IP and Destination IP Entropy values

The combination of source port, destination port, and destination IP entropies shows visible clusters; which can be attributed to different anomalous events. In Figure 24, cluster 1 includes a region having entropy values of 0 to 2.8 for all three features. The second cluster represents the scanning by the Honeypot for different IRC channels. This is based on the entropy values and the manual analysis of the trace. In this region, both the source port entropy and the destination IP entropy are high as the Honeypot is scanning for different IP addresses. The third cluster includes a region where there were bruteforce attempts to log into the SSH service running on the Honeypot. In this region, the source port entropy is high and the destination port entropy is low as these attacks are targeting the SSH port. The fourth cluster indicates a network scan performed by the Honeypot; which scans the SSH port on the destination machines using different ports for each connection. The region closer to zero mostly represents the IRC traffic as there are few machines communicating with each other using the IRC ports.

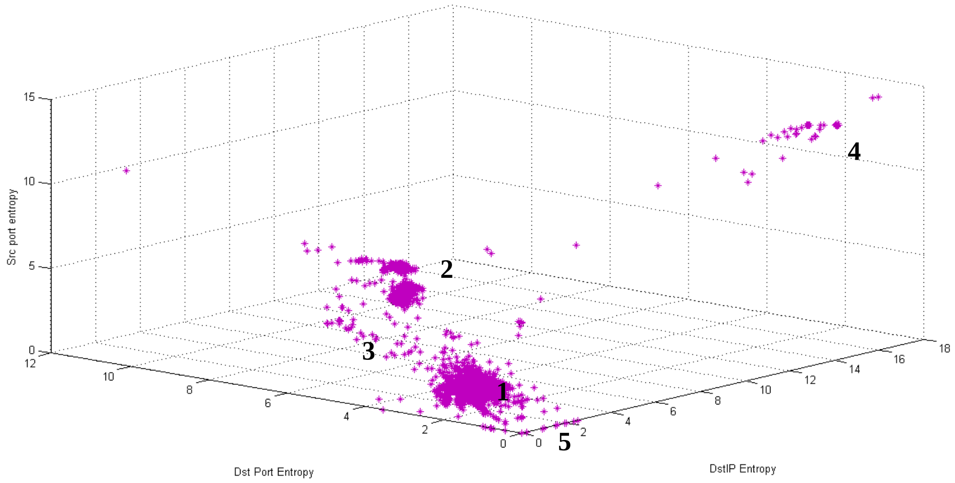


Figure 24: Combination of Destination IP, Destination Port, Source port entropy values

The actual behavior of different anomalies is explained in the following section. Table 9 summarizes the findings of feature analysis by providing the detection capabilities of various features.

After testing various combinations of traffic features, we conclude that the combination of destination port entropy, source port entropy, and destination IP entropy provide better detection capabilities. On the other hand, the volume features, i.e., total payload bytes and total packets have better detection capabilities and are very useful in detecting certain types of anomalies; which are not detected by traffic features. For example, certain malicious files transferred to the Honeypot were not detected by the feature-based parameters; while volume-based parameters detected these events. Therefore, instead of just using the feature-based techniques, we also need to use the volume-based techniques in order to detect most types of anomalies in a Honeynet.

Table 9: Summary of Detection Capabilities of various features

Traffic Feature	Detection Capabilities
Packet Size Entropy	Shows good variations but does not help in understanding the anomaly.
Destination IP Entropy	Shows large variations during specific anomalies and gives a good indication of an anomaly.
Source IP Entropy	Shows less variations in the traffic compared to the destination IP entropy.
Destination Port Entropy	Shows large variations for various anomalies.
Source Port Entropy	Shows large variations for various anomalies.

Traffic Feature	Detection Capabilities
Average Packet Inter-Arrival Time	Shows good variations but not very useful in understanding the anomaly behavior.
Total Payload Bytes	Shows good variations during most of the anomalies and when used with other features gives good understanding of the anomaly.
Total Packets	Shows good variations during anomalies and very useful in understanding the anomalies.
Average Payload Size	Shows good variations during anomalies but does not aid in understanding the anomaly behavior.

5.2.3. Malicious Activities Classification

This section describes the method used to classify different malicious activities using the selected features. The first step is to define the threshold levels for the selected features and then use these threshold levels to identify the behavior pattern of different anomalies.

5.2.3.1. Defining Thresholds for Different Features

In our proposed approach, anomalies are identified using the five top-ranked features: Destination IP entropy (DIP), Destination Port entropy (DP), Source Port entropy (SP), Total Payload Bytes (TB) and Total Packet Count (PC). The classification between normal and abnormal traffic is performed using the entropy and volume variations of the corresponding features. For example, the sample instances (which represent rows in Table 10 and Table 11) taken from Honeynet data collected from different sources indicate that during normal behavior very few variations in either entropy or volume values are seen, as shown in Table 10. However, there are significant traffic changes during the presence of anomalies or malicious activities, as shown in Table 11. Based on a thorough manual analysis of the training data sets, we found that during normal traffic, i.e., traffic that is not part of malicious activities, the entropy based features had an entropy variation in the range of 0 to 3. Similarly, for volume-based features, variations of normal traffic were in the range of 0 to 3000 bytes for the total payload bytes and 0 to 50 packets for the total packet count.

Table 10: Entropy and Volume Values for Normal Traffic

DIP	DP	SP	TB	PC
0	1.31	1.31	228	6
1	1.52	0.98	444	4
0	1.87	2.04	2631	20
1	0.918	1.58	3	1
0	1.62	0.33	168	8

DIP - Destination IP entropy

DP - Destination Port entropy

SP- Source Port entropy

TB - Total Payload Bytes

PC - Total Packet Count

Table 11: Entropy and Volume Values for Abnormal Traffic

DIP	DP	SP	TB	PC
0.52	3.56	4.46	12118	152
0	3.22	4.20	13971	138
0	3.37	3.61	70497	185
17.14	0.67	14.33	141048	5702
16.87	0.677	14.36	181988	7023
0.419	11.55	11.53	374099	4152
0.218	12.26	12.26	214096	5374

Based on the various entropy and volume variations seen in the Honeynet data, both during normal and anomalous traffic, threshold levels can be defined to distinguish between normal and abnormal traffic regions. The analysis of the entropy and volume changes recorded for the anomalies present in the datasets used (see Table 5) shows that entropy values greater than 3 are considered anomalous as indicated by the values reported in Table 10 and Table 11. Similarly, a volume change of the total bytes that is greater than 3KB or that of the total packets that is greater than 50 are considered malicious. These values are initially used to classify Honeynet traffic into normal and abnormal regions, i.e., detection of an anomaly. In addition, various other threshold levels are defined based on the entropy values and volume changes to identify the different types of malicious activities. These levels were obtained by analyzing the entropy and volume values of anomalous traffic in many traces, including those presented in Table 11. The behavior of different types of malicious activities can then be identified by the selected features and the associated threshold levels. For the purpose of easy mapping between types of malicious activities and threshold values, we define the following threshold levels:

- **Very High Entropy or Very High Volume:** This level is used for high entropy values and high volume of data. Based on the tests made on the traces only few anomalies, i.e., network scan and port scan, had high entropy values. The entropy values greater than 7 are considered as very high. Volume changes greater than 500KB and packet count greater than 2000 packets are also considered very high.

- **High Entropy and High Volume:** This level is used for entropy values that lie between 5 and 7. Based on the experimental results, it can be understood that certain anomaly types such as bruteforce attacks or fuzzers result in high entropy values. The reason for certain anomalies to have high entropy is due to the fact that they initiate too many connections from different ports to crack the passwords or exploit the vulnerabilities of different applications. Volume changes between 50KB and 500KB as well as packet count between 500 to 2000 packets are considered high.

- **Medium Entropy and Medium Volume:** This is used for entropy values that are greater than the normal range and less than the high entropy values. The entropy values that lie between 3 and 5 are considered medium. Most of the anomalies lie in this range as they cause enough changes in the entropy values to cross the normal range. The reason for this is that most of the anomalies target specific ports and do not require port scans, and hence the entropy values are slightly less compared to high entropy values. Volume changes between 3KB and 50KB as well as packet count between 50 to 500 packets are considered medium.

- **Zero entropy value:** This entropy value is used for cases during which only one dominant feature value is present in the trace. For example, if only one destination IP is visible during the five minute interval, then an entropy value of zero is recorded. This level is used only for feature-based parameters and is not applicable to volume-based parameters. Also, the situation in which this level is considered an anomaly is when there is zero entropy for the three feature-based parameters and a medium volume change.

Table 12 summarizes the various levels used to identify the malicious activities' behaviors in the Honeynet traffic.

Table 12: Threshold Levels used for Identifying Malicious Activities in Honeynets

Threshold Level	Range
Very High Entropy and Very High Volume	Entropy > 7 Bytes > 500Kb Packet Count > 2000
High Entropy and High Volume	7 > Entropy > 5 500kb > Bytes > 50Kb 2000 > Packet Count > 500
Medium Entropy and Medium Volume	5 > Entropy > 3 50Kb > Bytes > 3Kb 500 > Packet Count > 50
Zero Entropy	Zero Entropy (Also there should be medium volume changes)

5.2.3.2. Classifying Malicious Activities based on Features' Thresholds

It is essential to analyze the behavior of the various malicious activities detected in the training data sets after having already defined the required features to detect such anomalous activities. This analysis of the behavior of the various malicious activities detected will help in recommending a classification of such activities based on their behavior pattern. Hence, this section presents the various entropy and volume ranges, i.e., threshold levels, that were recorded for different types of malicious activities found in the training data sets. These ranges were used to learn the behavior pattern of different malicious activities and to recommend a mapping between the features' thresholds and the types of such activities, hence providing a classification of malicious activities based on the different features' threshold levels. The behavior pattern of each type of a malicious activity is defined using the five features that were selected earlier for anomaly detection.

Based on the analysis of the various training data sets that are presented later, it was found that not all the features are required to define the behavior of all the malicious activities.

Certain malicious activities can be defined using just two or three features while others require all the features. The reason for this is that certain malicious activities such as ICMP flood are independent of specific features such as port entropies that do not pertain to such malicious activities. Accordingly, certain features have values in the normal range in all instances of the same malicious activity in different training data sets due to which they do not aid in identifying such malicious activity.

The following set of tables summarizes the analysis of the behavior of all the malicious activities based on the various training data sets. It should be noted that the feature that was considered less important to define the behavior of the malicious activity is grayed out in the corresponding tables. The values recorded for the system compromise event from the different training data sets is shown in Table 13.

Table 13: Malicious Activity Type: System Compromise

Training Data set	Dst IP entropy	Dst Port Entropy	Source Port entropy	Total Payload Bytes	Total Packet Count
Scan 28	0 to 2.2	2.02 to 2.988	2.02 to 3.11	4547 to 742346	22 to 1491
Scan 14	0 to 1.84	3.15 to 3.56	3.065 to 4.465	12118 to 70497	138 to 185
Scan 19	0.9893	1.8078	2.159	1191 to 13145	33 to 102
SSH-based Honeypot	1.222305282	2.089387035	2.077343541	343184	385

Based on the recorded values, it can be concluded that the behavior of the system compromise malicious activity is Medium Destination Port Entropy, Medium Source Port Entropy, High Total Payload Bytes, and Medium Total Packet Count. In this case, the destination IP entropy is less significant because during the system compromise there is only one target machine being exploited and hence there is no significant change in the destination IP entropy values.

Table 14 shows the values recorded for malicious file downloads in different training data sets. Based on these values, the behavior of malicious file download can be defined as Very High Total Packet Bytes and High Packet Count. Note that the entropy values are omitted from Table 14 as they did not show any significant changes in the different training data sets and were in the normal range. The reason for this is that during a malicious file download, there is no significant change in the entropy values since most of the communication occurs between two machines using specific ports, i.e., FTP, HTTP, etc.

Table 14: Malicious Activity Type: Malicious File Download

Training Data set	Dst IP entropy	Dst Port Entropy	Source Port entropy	Total Payload Bytes	Total Packet Count
Scan 28		X	X	392336 to 42346	753 to 1491
Scan 14		X	X	16805 to 70497	145 to 185
Scan 19		X	X	374099	4152
SSH-based Honeypot		X	X	103512 to 1271603	1727

Table 15 shows the values recorded during the IRC communications that were noticed in the different training data sets. Based on these values, the behavior of the IRC communications can be defined as Zero Destination IP entropy, Zero Destination Port entropy, Zero Source Port entropy, Medium Total Payload bytes, and Medium Total Packet Count.

Table 15: Malicious Activity Type: IRC communication

Training Data Set	Dst IP entropy	Dst Port Entropy	Source Port entropy	Total Payload Bytes	Total Packet Count
Scan 28 Day 1	0 to 2.5	Many 0 points 1 to 2.5	Many 0 points 1 to 2.6	6200 to 19048	10 to 97
Scan 28 Day 3	0	0	0	1657 to 8652	15 to 75
SSH based Honeypot	1.58	1.79	1.78	26263 to 10660	229 to 249

Table 16 shows the various values recorded during the ICMP flood anomaly. The values indicate this malicious activity behavior as High Total Payload Bytes and Medium Total Packet Count. The reason that this malicious activity does not cause any changes to port entropies is that ICMP is a layer 3 protocol and does not include the ports that are used by the layer 4 protocols.

Table 16: Malicious Activity Type: ICMP flood

Training Data set	Dst IP entropy	Dst Port Entropy	Source Port entropy	Total Payload Bytes	Total Packet Count
Scan 28	0.721 to 3.4	0 to 1.38	0 to 1.63	6348 to 16177	6 to 58
SSH based Honeypot	1.584	0	0	14372 to 56636	14 to 55

Table 17 shows the values recorded during the port scan malicious activity. Based on these values, the behavior of port scan malicious activity can be defined as Very High Destination Port Entropy, Very High Source Port entropy, High Total Payload Bytes, and Very High Packet Count. Since this malicious activity basically scans the ports on the target machine, it is independent from the Destination IP entropy.

Table 17: Malicious Activity Type: Port Scan

Training Data set	Dst IP entropy	Dst Port Entropy	Source Port entropy	Total Payload Bytes	Total Packet Count
Scan 28 Day 1	0 to 0.91	7.09 to 8.685	6.95 to 9.81	153112 to 764302	406 to 3197
Scan 28 Day 3	0 to 0.39	4.99 to 7.424	5.29 to 9.61	56066 to 169238	674 to 2773
Scan 19	0.218 to 0.419	11.5 to 12.263	11.53 to 12.26	214096	5374
SSH based Honeypot	2.00 to 4.91	4.51 to 5.877	3.66 to 5.94	15289	154

Table 18 shows the variation of different features during the network scan malicious activity. Based on the recorded values, the network scan behavior can be defined as Very High Destination IP entropy, Very High source Port Entropy, High Total Payload Bytes, and Very High Total Packet Count. We should note that the network scan involves the scanning of a large number of IP addresses, and therefore, it is independent of the Destination Port entropy.

Table 18: Malicious Activity Type: Network Scan

Training Data set	Dst IP entropy	Dst Port Entropy	Source Port entropy	Total Payload Bytes	Total Packet Count
SSH based Honeypot	16.87 to 17.14	0.037 to 0.67	10.97 to 14.3	117906 to 10677114	1603 to 163519

Table 19 shows the variation of the different parameters recorded during a bruteforce malicious activity. Based on these values, the behavior of a bruteforce malicious activity can be defined as Medium Destination port entropy, High Source Port Entropy, Medium Total Payload Bytes, and High Total packet count. During bruteforce attempts, most of the communication occurs between two machines and hence it does not cause significant changes in the destination IP entropy.

Table 19: Malicious Activity Type: Bruteforce

Training Data set	Dst IP entropy	Dst Port Entropy	Source Port entropy	Total Payload Bytes	Total Packet Count
SSH based Honeypot	0 to 1.4	0 to 4.39	3.98 to 6.53	29680 to 76402	494 to 1947

As we can see from the previous analysis, we have used different number of instances to identify the different types of malicious activities. It can also be stated that the malicious activities' behaviors that were identified based on more than one training data set have more significance compared to the behaviors identified based on only one training data set. Hence, the presence of more instances of the same malicious activity in different traces will be useful in accurately predicting other similar types of malicious activities' behaviors.

Table 20 lists a classification of the various malicious activities and their associated behavior in terms of different features. Identifying the behavior of different malicious activities will help in detecting similar activities in other data sets. Using a large number of data sets will help in defining the behavior of the malicious activities better. This information can then be used to detect similar malicious activities by comparing the detected behavior to the proposed classification.

Table 20: Classification of the Behavior of Different Malicious Activities

Anomaly	Dst IP entropy	Dst Port Entropy	Source Port entropy	Total Payload Bytes	Total Packet Count
System Compromise	▨	M	M	H	M
Malicious File Download	▨	▨	▨	VH	H
IRC communications	Z	Z	Z	M	M
ICMP flood	▨	▨	▨	H	M
Port Scan	▨	VH	VH	H	VH
Network Scan	VH	▨	VH	H	VH
BruteForce	▨	M	H	M	H

VH: Very High, **H**: High, **M**: Medium, **Z**: Zero

5.2.4. Results of Malicious Activities Classification

The evaluation of the proposed technique was carried out by developing a Java code for extracting entropy values from the traces. The jNetPcap Java API was used for developing the code to read the PCAP trace files and then the entropy values for every five minute time interval was calculated for different features. The results were then plotted using the five top-ranked features. The trace files used for obtaining the results are:

- Scan 27: Honeynet.org Scan of the Month Challenge, March 2003.
- Lab Trace with Synthetic Anomalies
- Dionaea capture trace: this trace was collected using a local installation of a Dionaea low interaction Honeypot.

The results are comprised of anomalies that were identified using the proposed technique, as well as the detection rate and corresponding plots. The main plots that are presented in the results are the 3-D cluster plot and the two volume feature plots. The three features used for the 3-D plots are the Destination IP Entropy, Destination Port Entropy, and Source Port Entropy. The features used for volume feature plots are the Total Payload Bytes and Total

Packet Count. In addition to this, the time-based view of the cluster plot was also used to visualize the spread of anomalies.

The results mainly focus on the detection rate obtained using the proposed technique. The detection rate is calculated by comparing the number of anomalous events reported by our technique (including scanning, system compromise, malwares, rootkits downloads, etc.) to the number of anomalous events present in the trace. In Honeynet systems, all traffic coming to the Honeypots is considered malicious. Based on this fact, we are not presenting the false alarm rate and we consider that anomalies that are classified as belonging to a given type are all malicious in nature. The main reason for this assumption is that most of the traffic that arrives on a Honeynet is by default malicious in nature. Apart from this, a Honeypot also receives packets from other network devices which may not be always malicious, such as network discovery packets coming from windows based machines. In this work, the main purpose is to locate where anomalies occur in very large Honeynet traces, which is a tedious task if done manually. Based on our experiments, we also found that significant changes in Honeynet traffic occurred only during malicious events, which essentially serves to locate points of anomalous activity within a given traffic profile.

5.2.4.1. Scan 27 Trace

The main plots that were used to classify anomalies are the cluster plots and volume-based parameters plots. The three feature cluster plot is shown in Figure 25. The entropy points that are above the threshold level of 3 are related to malicious activities seen in the scan 27 trace.

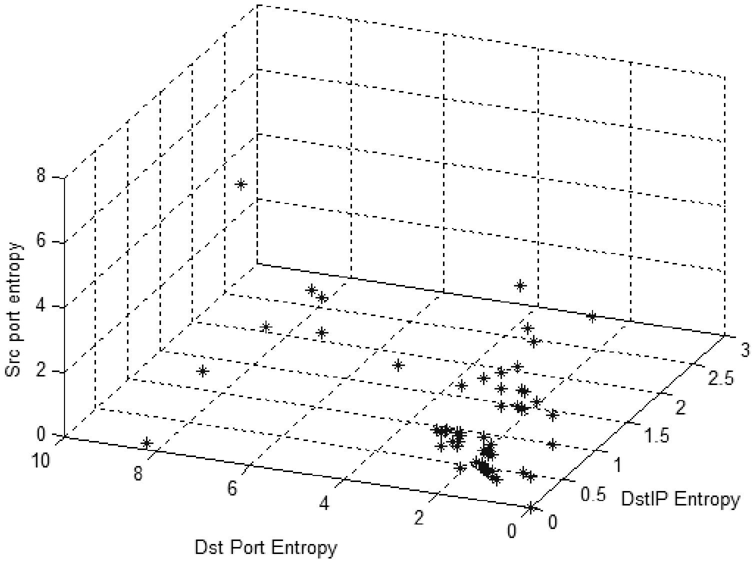


Figure 25: Cluster plot for the Scan 27 trace

The volume plot shown in Figure 26 also shows some peaks from the end of the third day indicating some malicious activity.

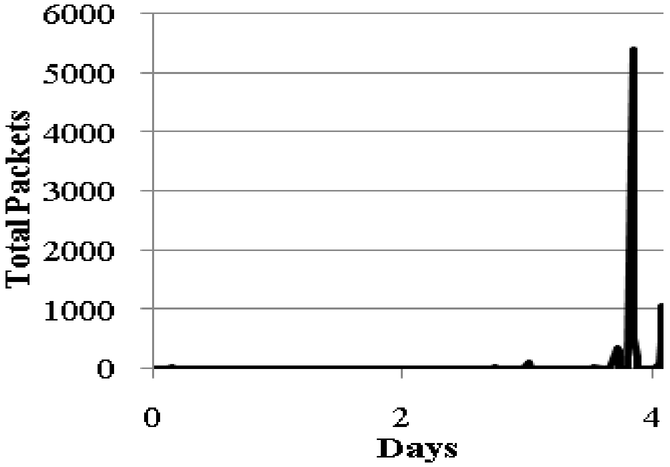


Figure 26: Total Packet Count for the Scan 27 trace

Based on successful classification of anomalies, the analysis results of the scan 27 trace are listed in Table 21. A total of 11 anomalous events which fall in the categories mentioned in Table 21, were detected using the proposed technique. Honeynet.org reported 12 anomalous events during this scan of the month challenge. The anomaly that was undetected by our proposed technique represents attacks on the MS-SQL server UDP port 1434 which resembled the slammer worm. A total of 55 packets were sent to the Honeypot targeting port 1434 but these packets were sent at different times during the five day period. The reason for this remaining undetected was that the time gap between these packets was large and did not cause any noticeable and rapid change in entropy. Based on the results, a detection rate of 91.6% was achieved using the proposed technique. The time view plot in Figure 27 displays the spread of events throughout the five day period. The different colors represent different times. Events with the same color in the plot indicate that they happen in the same time period. This allows us to order the different events based on their time of happening, which provides an understanding of the sequence of malicious activities. This could provide more insight into the tactics used by the attackers targeting Honeynets. It is clear from this plot that a system compromise attempt happened on the third day while the port scan and other anomalies occurred on the fourth day.

Table 21: Anomalies Detected in Scan 27 Trace

Detected Behavior	Type of Anomaly (by Manual Analysis)	Actual number of Occurrences Reported
TB(M), PC(M) (5 occurrences)	SMB attacks	5 occurrences reported
DP(M), SP(M), TB(H), PC(M) 1 occurrence	Buffer overflow attempt and System Compromise	1 occurrence reported
TB(VH), PC(H) 1 occurrence	Malicious file download	1 occurrence
DP (VH), SP(VH), TB(M), PC(M) 1 occurrence	Port Scan.	1 occurrence
DP(M), SP(M), TB(H), PC(H) 1 occurrence	HTML script kiddies	1 occurrence
TB(M) 1 occurrence	Attempts to exploit buffer overflow	1 occurrence CodeRedII worm
DIP (Z) , DP(Z), SP(Z), TB(M) PC(M) 1 occurrence	IRC communications	1 occurrence
Not Detected	X	Slammer Worm, 1 occurrence
Total Detected = 11		Total Reported = 12
Detection Rate = 91.6%		

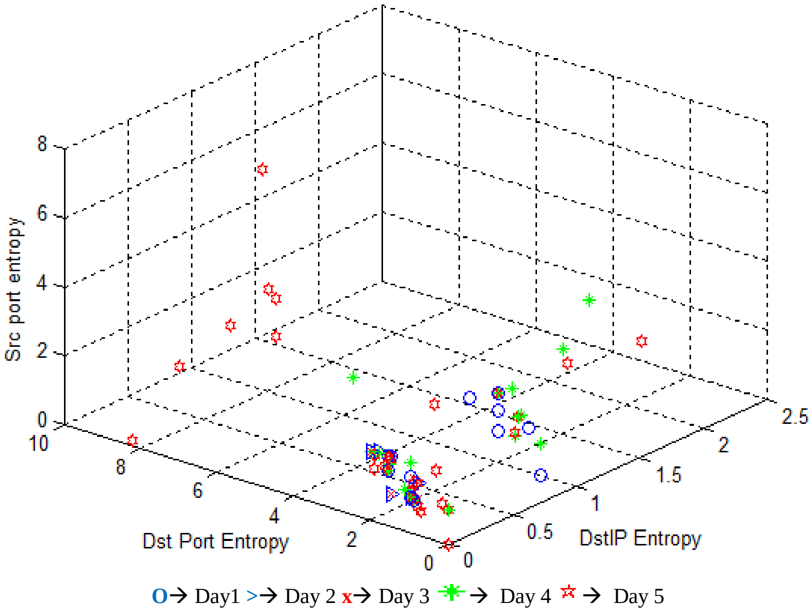


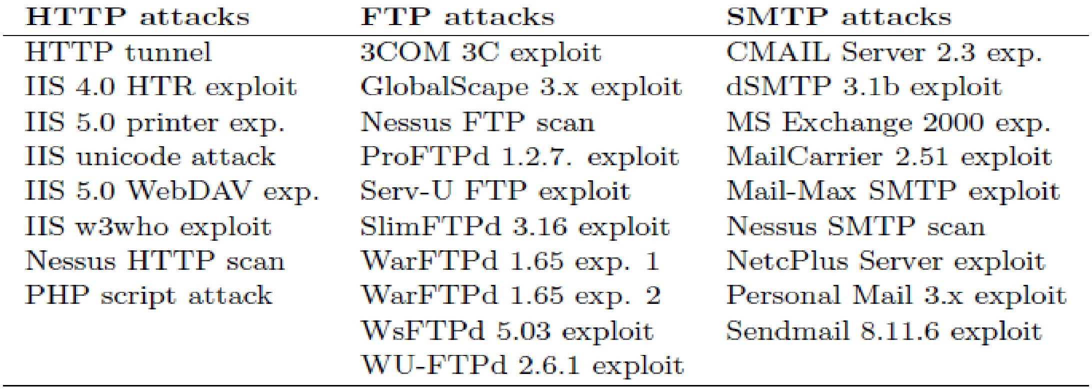
Figure 27: Time View Plot for Scan 27 trace

5.2.4.2. Lab Trace with Synthetic Anomalies

Lab trace used for evaluating the detection technique was generated in the Lab setup within KFUPM. A Honeynet was setup with Honeywall - a high interaction Honeypot and Windows XP Honeypot. The BackTrack 4.1 operating system was used as the attacker machine, which was used to attack the windows XP Honeypot with different types of attacks. The Honeypot was made visible on the network and popular services were activated on it such as IIS web server, FTP server, SSH server etc. The main tools that were used from the BackTrack operating system were:

- Nmap
- Open VAS vulnerability scanner
- Metasploit Penetration Testing Framework 3.0

Metasploit Framework [38] is one of the most popular open source penetration testing tools that are available in the market [39]. We used these tools to generate a trace that includes different types of malicious activities and then used our technique to test whether it can detect these anomalies. Metasploit framework has been used by other authors to generate a similar data set for their anomaly detection techniques. Laskov and Kloft [40] have used the metasploit framework to create a malicious dataset by generating various exploits from the tool. Rieck, and Laskov [41] have also used the metasploit framework to create a malicious dataset. They used various exploits from this framework which are shown in Figure 28. Düssel et al [42] also used the metasploit framework to generate malicious dataset for testing their anomaly detection technique.



HTTP attacks	FTP attacks	SMTP attacks
HTTP tunnel	3COM 3C exploit	CMAIL Server 2.3 exp.
IIS 4.0 HTR exploit	GlobalScape 3.x exploit	dSMTP 3.1b exploit
IIS 5.0 printer exp.	Nessus FTP scan	MS Exchange 2000 exp.
IIS unicode attack	ProFTPd 1.2.7. exploit	MailCarrier 2.51 exploit
IIS 5.0 WebDAV exp.	Serv-U FTP exploit	Mail-Max SMTP exploit
IIS w3who exploit	SlimFTPd 3.16 exploit	Nessus SMTP scan
Nessus HTTP scan	WarFTPd 1.65 exp. 1	NetcPlus Server exploit
PHP script attack	WarFTPd 1.65 exp. 2	Personal Mail 3.x exploit
	WsFTPd 5.03 exploit	Sendmail 8.11.6 exploit
	WU-FTPd 2.6.1 exploit	

Figure 28: Exploits used for generating malicious dataset [41]

The attacks that were generated in our experiment are listed in Table 22.

Table 22: Attacks Generated Against the Honeypot

Categories	Types of Attacks
Port Scan	NMAP regular scan NMAP quick scan NMAP intense scan NMAP slow comprehensive scan
Vulnerability Scanning	Open VAS Scanner
Database attacks	MYSQL login utility scanner MYSQL database access attempts
Server Message Block (SMB) protocol attacks	SMB Negotiate Dialect Corruption (Fuzzers/smb/smb_negotiate_corrupt) Microsoft Workstation Service NetAddAlternateComputerName Overflow Microsoft Server Service Relative Path Stack Corruption Microsoft Server Service NetpwPathCanonicalize Overflow Microsoft Plug and Play Service Overflow Microsoft Print Spooler Service Impersonation Vulnerability
DCE/RPC, (Distributed Computing Environment / Remote Procedure Calls) attacks	Endpoint Mapper Service Discovery (scanner/dcerpc/endpoint_mapper) DCERPC TCP Service Auditor Microsoft RPC DCOM Interface Overflow exploit Microsoft Message Queueing Service Path Overflow exploit
FTP	Simple FTP Fuzzer FTP attack access gain attempt
HTTP IIS web server attacks	Microsoft IIS WebDAV Writ exploit Microsoft IIS 5.0 Printer exploit Microsoft IIS/PWS CGI Fil exploit Microsoft IIS 5.0 WebDAV ntdll.dll Path Overflow
SMTP attacks	MS03-046 Exchange 2000 XEXCH50 Heap Overflow exploit
SNMP attacks	Network Node Manager Snmp.exe CGI Buffer Overflow
Backdoor	Energizer DUO Trojan Code Execution
SSH attacks	SSH Key Exchange Init Corruption

The lab trace was used to test the effectiveness of the proposed technique. In this trace synthetic anomalies were injected using various tools used for penetration testing. The most popular tools that were used are: NMAP, OpenVAS scanner and Metasploit. The Metasploit tool was used to generate system exploits which target various services on the Honeypot. The attacks were generated five days after the Honeypot was connected to the Internet. In this lab trace, 27 anomalies were inserted (refer to Table 22 for the list of inserted anomalies). But, as this Honeypot was connected to the Internet other attacks were also detected. The cluster plot in Figure 29 shows various groups of anomalous activities.

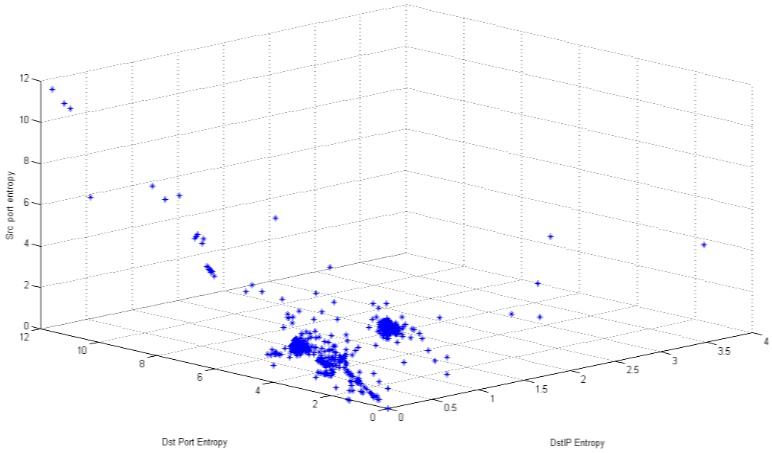


Figure 29: Cluster view of Labtrace

The volume based parameters also show significant changes during the anomalies as seen in Figure 30 and Figure 31. The peaks correspond to scanners and malicious data transfers. Table 23 shows the categories of anomalies, i.e., malicious activities, detected in LabTrace.

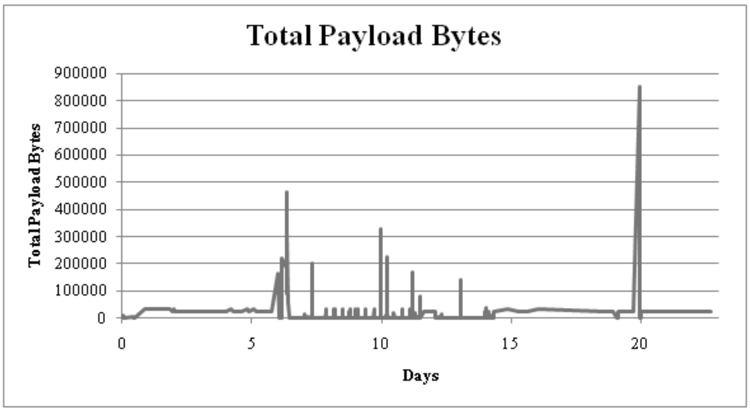


Figure 30: Total Payload Bytes in Labtrace

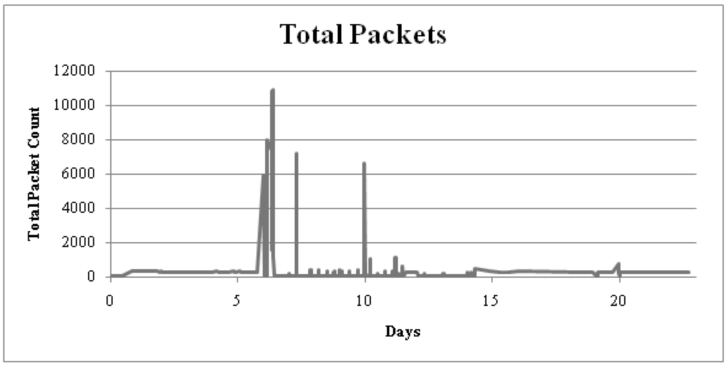


Figure 31: Packet Count for Labtrace

Table 23: Categories of Anomalies Detected in LabTrace

Detected Behavior	**Type of Anomaly by Manual Analysis**	**Anomaly Behavior by Reverse Mapping**	**Reported Number of Occurrences**
DP (VH), SP(VH), TB(H), PC(VH) 5 Occurrences	Port Scan Different NMAP port scan types were used to scan the Honeypot. The comprehensive scan shows the highest entropy value and the highest number of packets were sent in this type of scan.	Port Scan	5 occurrences
DP(VH), SP(VH), TB(H), PC (VH) 1 occurrence	Vulnerability Scanning Vulnerability scanning using Open VAS Scanner	Port Scan	1 occurrence
DP(M), SP(M), TB(H), PC(M) 12 occurrences	Vulnerability exploits (Metasploit exploits) These are system compromise attempts	System Compromise	12 occurrences
TB (H), PC(M) 3 occurrences	System Compromise	ICMP flood	3 occurrences
DP (M), SP(H), TB(M), PC(H) 4 occurrences	Password Brute force and Fuzzers These attacks used various combinations of username and passwords to guess the account password. The fuzzer tools fall in this category.	Bruteforce	4 occurrences
DP(M), TB(M), PC(M) 24 occurrence	SMB connection attempts Most of these attempts try to connect to the Microsoft-ds port (445) on the remote machine and try to gain access to the system shares.	Detected behavior not available in the known behavior set	X
Not Detected	X	X	SSH attack and Microsoft Message Queueing Service Path Overflow exploit 2 occurrences
Total Anomalies Detected = 25			**Total Anomalies Reported = 27**
Anomaly Detection Rate			**92.5%**

The major categories of anomalies that were detected in the Labtrace are listed in Table 23. A total of 27 attacks launched against the Honeypot using the Backtrack OS. Using the proposed technique, 25 attacks were successfully detected with a detection rate of 92.5%. The undetected anomalies were the SSH attack and the Microsoft Message Queueing Service Path Overflow exploit. The reason for not detecting these two attacks is that they did not cause significant changes in entropy values. In metasploit there was no exploit available for Open SSH (tool that was installed in the Honeypot) due this the exploit that was attempted did not succeed and only a few packets were launched during this attack. The second attack, i.e. Microsoft Message Queuing Service Path Overflow exploit was also not successful and hence it did not generate many packets to cause changes to the entropy values. A total number of 49 attacks which include the 25 attacks generated using metasploit and other attacks caused by other machines in the network were detected.

The K-means clustering was applied on the entropy values and Figure 32 shows the different clusters that were detected. Four cluster regions were detected in the LabTrace. The cluster one was the normal traffic and the cluster two and three represents metasploit exploits and SMB attacks, respectively.

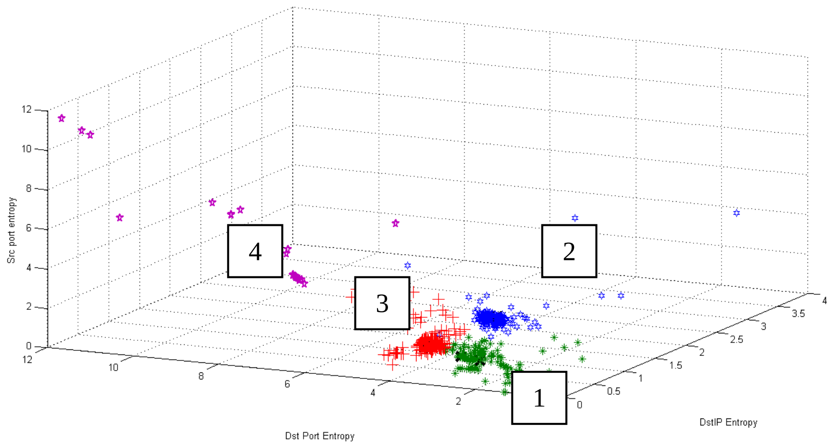


Figure 32: K-Means Cluster view of Labtrace

The time view trace (refer to Figure 33) shows the different times at which the events occurred in the Honeypot lab trace during the 25 day period. The attacks were generated five days after the Honeypot was connected to the Internet and most of the attacks were generated during the end.

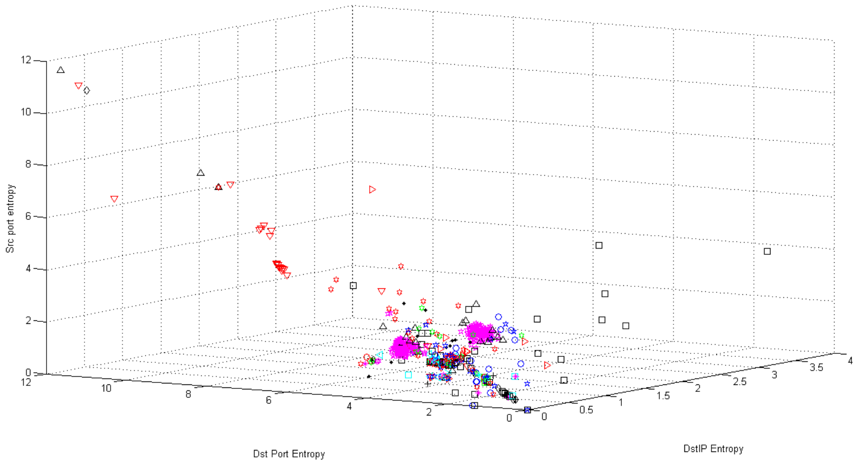


Figure 33: Time view of Labtrace

5.2.4.3. Dionaea Trace

The Dionaea trace was collected in the local University network. The cluster plot is shown in Figure 34. The points having high entropy values are related to the port scan activity seen in the trace. The points in the region of the Destination port entropy axis from 3 to 6 are related to brute force attempts and connection attempts to FTP and HTTP services.

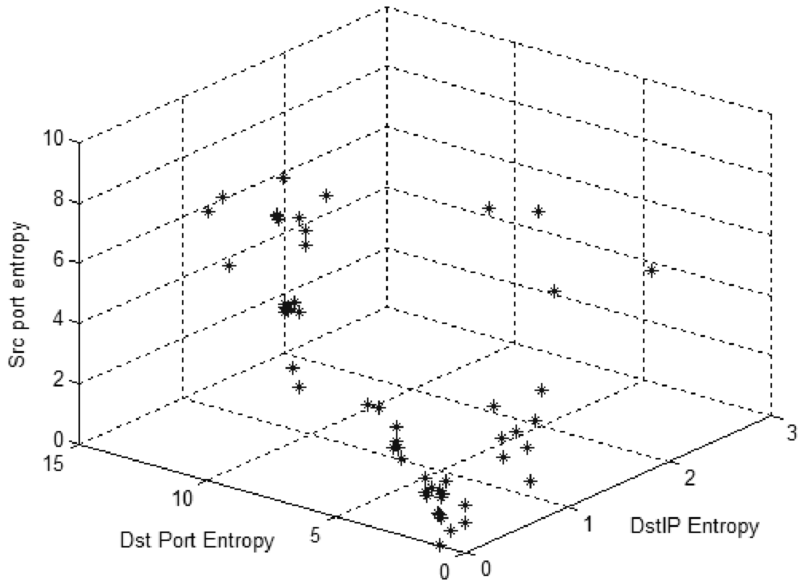


Figure 34: Cluster plot for Dionaea Trace

The volume plot in Figure 35 shows peaks in the initial period when many malicious activities were recorded.

Table 24 lists the anomalies that were detected in the Dionaea trace. A total of 10 anomalies were detected, i.e., 3 port scan attempts, one MSSQL brute-force login attempt, 2 scanning web robots, and 4 connection attempts to popular ports, as indicated in the categories listed in Table 24. This means a detection rate of 90.9% for the Dionaea trace. The SIP scanning worm was not detected using our proposed technique. This is due to the fact that during each connection attempt, it was sending only two packets to the Honeypot. Because of this, the anomaly did not cause significant changes in the selected features.

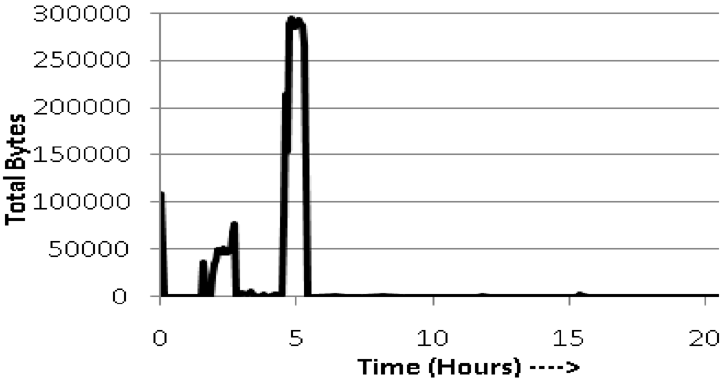


Figure 35: Total Payload Bytes for Dionaea Trace

Table 24: Anomalies Detected in Dionaea Trace

Detected Behavior	Type of Anomaly by Manual Analysis	Reported Anomalies
DP(VH), SP(VH), TB(H), PC(VH) 3 occurrences	Port Scan Different NMAP port scan types detected	3 occurrences
DP(M), SP(M), TB(H), PC(M) 1 occurrence	MS-SQL Brute force attempts Multiple login attempts were made to MSSQL server.	1 occurrence
DP(M), SP(H) 2 occurrences	Web Robots (also known as Web Wanderers, Crawlers, or Spiders)	2 occurrences
DP(M), SP(M) 4 occurrences	Connection attempts on popular ports (HTTP, FTP, MSSQL etc.)	4 occurrences
Not Detected	X	SIP worm reported 1 occurrence

Total Anomalies Identified= 10		Total Anomalies Reported = 11
Detection Rate = 90.9%		

The anomaly detection rate measures the percentage of anomalies detected based on all the anomalies reported for the trace. Combining the results of both traces, a total detection rate of 90.25% was achieved based on the threshold levels used in our proposed technique.

5.2.4.4. Recall and Precision of Anomaly Behavior Detection

Recall and precision metrics were used to evaluate the effectiveness of the reverse mapping applied to detect the anomaly behavior from the predicted behavior. Table 25 summarizes the anomaly detection rate as well as precision and recall percentages for the proposed technique. The metrics used are [43] :

$$Recall = \frac{True\ Posivites}{True\ Positives + False\ Negatives}$$

$$Precision = \frac{True\ Posivites}{True\ Positives + False\ Positives}$$

Table 25: Recall and Precision of Anomaly Behavior Detection

Data Set	**Anomaly Detection Rate**	**Anomaly Behavior Detection**				
		R	**D**	**I**	**Precision % (I/D)**	**Recall % (I/R)**
Scan 27 Trace	91.6%	10	4	3	75%	30%
LabTrace	92.5%	43	23	19	82.6%	44.18%
Dionaea Trace	86.6%	9	4	3	75%	33.33%
Total	**Anomaly detection Rate: 90.25%**	**62**	**31**	**25**	**80.65%**	**40.32%**

Table 26: Recall and Precision without considering multiple occurrences

Data Set	**R**	**D**	**I**	**Precision % (I/D)**	**Recall % (I/R)**
Scan 27 Trace	6	4	3	75%	50%
LabTrace	4	5	3	60%	75%

Dionaea Trace	3	2	1	50%	33.33%
Total	**13**	**11**	**7**	**63.63%**	**53.84%**

The anomaly detection rate measures the percentage of anomalies detected based on the reported behavior out of all anomalies reported for the trace. A detection rate of 90.25% was achieved based on the threshold levels used in the proposed technique. The effectiveness of mapping the detected anomaly behavior to the appropriate anomaly was found using precision and recall metrics. A precision of 80.65% and a recall percentage of 40.32% were achieved using the proposed technique when all the occurrences of different anomalies are taken into account. The precision and recall percentages for detecting the anomaly type i.e. without considering multiple occurrences for each type (Table 26) show better recall percentages compared to previous values in Table 25. The values obtained in Table 25 are based on the fact that certain anomalies occurred many times and did not match the predicted behavior set. This led to decrease of the recall value. The reason for getting low values for the recall is that we have only few number of anomaly behaviors mapped in the predicted behavior set. Therefore, we expect the recall to improve if more traces are used to update the mapping table obtained in the previous chapter.

5.2.5. Summary

In summary, we found that both feature-based and volume-based parameters are necessary to detect anomalies in Honeynet traffic. The proposed technique can be further validated using more data sets. The detection scheme was unable to detect stealth attacks or slow attacks as they do not cause clear variations to the entropy and volume features. This can be further improved by adding the capabilities of detecting such attacks.

6.0 CONCLUSIONS

The Saudi Honeynet Project (SAHNET) has successfully achieved its goal by setting a practical model comprised of a Honeynet network built within the KFUPM campus for collecting data about attackers' behavior as well as analyzing and assessing the security of the university's network. We have also proposed an extension of such architecture to the KSA level and that can be deployed with the support of the Saudi Computer Emergency Response Team (CERT-SA). The SAHNET project also contributed to the society by providing weekly security reports that summarized malicious and suspicious activities targeting KFUPM networks to CERT-SA.

In addition, the project led to the development of new techniques for the analysis of the Honeynet traffic, and thus automating and simplifying some of the tasks related to analyzing Honeynet traffic. Moreover, based on our research work, a number of publications have been produced and presented in reputable conferences. Through this project, we have also been able to establish our own Saudi Honeynet Chapter for the first time from the Kingdom, registered officially with the Global Honeynet Community. In addition, we have initiated collaborations with some major Honeynet chapters which we hope will result in valuable future research projects. Furthermore, the project strengthened the knowledge base of our team of researchers in this particular field.

Through the above achievements and contributions, we believe that the project significantly contributed to the ***computer systems and networks*** priority technology area of the Information Technology program of KSA's National Science, Technology and Innovation plan, and more specifically to the area of ***IT security and privacy***.

7.0 PROJECT OUTCOMES

Outputs	Status	Date
Publications (Journal Papers)		
1. Mohammed H. Sqalli, Syed Naeem Firdous, Khaled Salah, and Marwan Abu-Amara, "Classifying Malicious Activities in Honeynets using Entropy and Volume-based Thresholds", Security and Communication Networks.	**Accepted**	**17/8/2011**
2. Mohammed H. Sqalli, Syed Naeem Firdous, Zubair Baig, and Farag Azzedin, "Classification-based Identification of Malicious Activities in Honeynet Traffic".	**Under Preparation**	
Publications (Conference Papers)		
3. Zubair A. Baig, Saad Khan, Saif Ahmed, and Mohammed Sqalli, "A Selective Parameter-based Evolutionary Technique for Network Intrusion Detection", The 11th International Conference on Intelligent Systems Design and Applications (ISDA), Córdoba, Spain, November 22-24, 2011.	**Accepted**	**22-24/11/2011**
4. Mohammed H. Sqalli, Syed Naeem Firdous, Zubair A. Baig, and Farag Azzedin, "An Entropy and Volume-based Approach for Identifying Malicious Activities in Honeynet Traffic", International Conference on Cyberworlds, IEEE Computer Society, pp. 23-30, Banff, Alberta, Canada, October 4-6, 2011.	**Published**	**4-6/10/2011**
5. Mohammed H. Sqalli, Shoieb Arshad, Mohammad Khalaf, and Khaled Salah, "Identifying Scanning Activities in Honeynet Data using Data Mining", The 3rd International Conference on Computational Intelligence, Communication Systems and Networks (CICSyN 2011), pp. 178-183, Bali, Indonesia, July 26-28, 2011.	**Published**	**26-28/7/2011**

6. Mohammed H. Sqalli, Syed Naeem Firdous, Khaled Salah, and Marwan Abu-Amara, “Identifying Network Traffic Features Suitable for Honeynet Data Analysis,” The 24th Canadian Conference on Electrical and Computer Engineering, Niagara Falls, Ontario, Canada, May 8-11, 2011.	**Published**	**8-11/5/2011**
7. Syed Naeem Firdous and Mohammed H. Sqalli, “Identifying Features for Honeynet Data Analysis using Feature Evaluation,” Second Scientific Conference for Graduate and Undergraduate Students, Jeddah, 28-31 March 2011.	**Published**	**28-31/3/2011**
8. Mohammed H. Sqalli, Raed AlShaikh, and Ezzat Ahmed, “Towards Simulating a Virtual Distributed Honeynet at KFUPM: A Case Study,” The IEEE UKSim 4th European Modelling Symposium on Mathematical Modelling and Computer Simulation (EMS), Pisa, Italy, November 17-19, 2010.	**Published**	**17-19/11/2010**
9. Mohammed H. Sqalli, Raed AlShaikh, and Ezzat Ahmed, “A Distributed Honeynet at KFUPM: A Case Study”. The 13th International Symposium on Recent Advances in Intrusion Detection (RAID), LNCS 6307, pp. 486-487, Ottawa, Ontario, Canada, September 15-17, 2010.	**Published**	**15-17/9/2010**
10. Syed Naeem Firdous and Mohammed H. Sqalli, “Saudi Honeynet Project,” First Scientific Conference for Graduate and Undergraduate Students, Riyadh, 1-3 March 2010.	**Published**	**1-3/3/2010**
Presentations 1. Shoieb Arshad, “Identifying Scanning Activities in Honeynet Data using Data Mining”, The 3rd International Conference on Computational Intelligence, Communication Systems and Networks (CICSyN 2011), pp. 178-183, Bali, Indonesia, July 26-28, 2011.	**International Conference talk**	**27/7/2011**

2. Marwan Abu-Amara, “Identifying Network Traffic Features Suitable for Honeynet Data Analysis,” The 24th Canadian Conference on Electrical and Computer Engineering, Niagara Falls, Ontario, Canada, May 8-11, 2011.	**International Conference talk**	**9/5/2011**
3. Mohammed H. Sqalli, “Honeynet Traffic Analyzer Using Anomaly Detection Techniques”, the Annual Honeynet Project Workshop, ESIEA Institute, Paris, France, March 21-25, 2011.	**International Invited talk**	**23/3/2011**
4. Mohammed H. Sqalli, “Saudi Honeynet Project - Trapping The Hackers”, the Saudi Computer Emergency Response Team (CERT), Riyadh, Saudi Arabia.	**National Invited talk**	**16/3/2011**
5. Mohammed H. Sqalli, “Saudi Honeynet Project - Trapping The Hackers”, CCSE Seminar, KFUPM, Dhahran, Saudi Arabia.	**Institutional Seminar**	**4/1/2011**
6. Raed AlShaikh, “The Honeynet project at KFUPM: A Case Study”, KACST National Program for free/open source software technologies, Saudi Arabia.	**National Invited talk**	**19/7/2010**
7. Syed Naeem Firdous, “Saudi Honeynet Project,” First Scientific Conference for Graduate and Undergraduate Students, Riyadh, 1-3 March 2010.	**National Conference talk**	**2/3/2010**
Technical Outputs List any technical outputs such as software programs, databases, algorithms, and measurement instruments. 1. We have setup of an initial Honeynet in the security lab within the CCSE College. And, on March 28, 2010, the Honeynet deployment of the Saudi Honeynet Project was successful in detecting the first worm, which was a blaster worm evident through the log files. 2. Many Honeypots were deployed within the campus which recorded different types of attacks targeted to the KFUPM network. Many different malware attacks were seen such as attacks on SMB, MS-SQL, etc.		

Many new tools were tested and used to analyze and understand the behavior of these attacks. 3. A new data analysis approach was proposed for quickly analyzing the Honeynet traffic to identify the anomalies. 4. A tool is being finalized and will be shared with the worldwide Honeynet Project community for automating the analysis of Honeynet data.		
Patents, licenses or other research commercialization activities		
Other List any other forms of research dissemination that is intended for non-scientific audiences (such as radio talks, newspaper articles, television appearances). 1. We have launched a web site for the project on June 19th, 2010, and which can be found at: www.kfupm.edu.sa/Honeynet 2. We initiated collaboration with Saudi Computer Emergency Response Team (CERT), Malaysian CERT, and UAE CERT. 3. We are sending weekly reports to CERT-SA about the malicious activities collected by our Honeynets.		

RELATIONSHIP OF THE PROJECT OUTCOMES TO NSTIP STRATEGIC FRAMEWORK

PROJECT OUTCOMES	STRATEGIC TECHNOLOGY PROGRAM GOALS			PROJECT OBJECTIVE ACHIEVED
	GOAL 1 (Computer prototypes, systems, executable product(s), process(es), or procedure(s) useful to the local industry and relevant to the strategic technologies roadmap of IT program tracks)	**GOAL 2** (Experimental setups and equipments contributing to building computer prototypes/systems, executable product(s), process(es), or procedure(s) in IT program tracks)	**GOAL 3** (Experienced teams in the technologies related to the projects)	
1. A prototype of a Honeynet that is able to capture, collect, report, and analyze network attacks	**X**			**1, 2, 4**
2. A KFUPM Honeynet that is able to assess the health of the KFUPM campus networks	**X**			**2, 3, 4, 5**
3. Captured data should be readily available for further analysis by researches to discover new type of attacks		**X**		**2, 4**
4. The suitability of the wide variety of software tools for collection, analysis, statistics reporting should be studied			**X**	**1, 2, 4**
5. A web site that reports periodically (per hour or per day) online statistics and charts on different network attacks and worms	**X**			**6, 7**
6. A complete report on findings, recommendation, and learned lessons	**X**			**3, 5**
7. A design recommendation of the most suitable Honeynet architecture comprising a number of Honeypots to be distributed and widely deployed at different locations of the KSA Internet		**X**		**3**
8. Strong collaborations with well-known researchers throughout the globe who are part of the Honeynet alliance			**X**	**6, 7**
9. The project will institute independent tools and products to diagnose, secure, and protect KSA networks and Internet			**X**	**3**
10. The project will be a great benefit to Saudi schools, universities, private		**X**		**5**

companies, etc.				
11. The project will enable building local expertise and knowledge-base in installing, integrating and developing Honeynets in KSA			**X**	**4**
12. The project will provide a framework for the dissemination of valuable findings among KSA government organizations and corporate sector thus contributing to improve confidence in the security of KSA networks	**X**			**6**
13. The project could lead to the development of a center of excellence or consultancy for other local government and private organizations for the research and development as well as deployment of Honeynets in KSA			**X**	**7**

8.0 ADDITIONAL ACHIEVEMENTS

The following are additional achievements other than the objectives set in the proposal:

1. The Saudi Honeynet Project Chapter met all the requirements and became the 2nd Arab Honeynet Project Chapter in the world on July 26th, 2010, and it is one among 40 chapters worldwide. Since then, we have been actively involved with the worldwide Honeynet project since we became an official chapter. The list of official chapters can be found here: http://www.Honeynet.org/og
2. We have organized seminars and workshops at KFUPM on topics related to web security, Android malware, malware evolution, and PDF attacks; delivered by two Malaysian Cybersecurity experts in June 4-8, 2011. More than 30 faculty, staff, and students attended these events. The details of this program can be found in Appendix 4. This allowed for the transfer of knowledge to the local community as well.
3. The project resulted in one completed master thesis and at least one additional ongoing master thesis. The completed thesis resulted in a number of journal and conference publications as outlined in section 7.
4. We have established a Honeynet lab where many students are trained on how to design, deploy, and use Honeynets.
5. SANS, one of the most trusted and largest source for computer, network, and information security training and research in the world, tasked members of the Saudi Honeynet (SAHNET) Project at KFUPM to be responsible for the Arabic version of OUCH!, which is the SANS free, monthly security awareness newsletter. This will start with the May 2011 newsletter.

9.0 VALUE TO THE KINGDOM

The ultimate goal of this project has been to study ways to enhance the security of governmental, industrial, and public networks within KSA. The studied and deployed Honeynet allowed us to provide information surrounding security threats and vulnerabilities active in the wild on KSA networks today, and to learn the tools, tactics, and motives of the blackhat community. Based on the Honeynet data collected and the reports generated, it is now possible to identify cyber-attacks being launched inside KSA and also launched from outside KSA and targeted towards KSA based on the data collected and the reports generated. Clearly, the project has contributed to achieving the objectives of the national plan of science and technology by providing an infrastructure and a working model to assess the security health and better understand the vulnerabilities that can possible exist in different types of KSA networks, thereby finding appropriate means and solutions to secure these networks against insides and outside attacks.

The project allowed for the dissemination of valuable findings to the Computer Emergency Response Team in Saudi Arabia (CERT-SA). CERT-SA uses considerably benefit from our Honeynet collected data. Weekly reports summarizing all the findings including attackers' activities are shared with CERT-SA for the purpose of harnessing our local observations and findings with global reports generated at CERT-SA, to improve network security of the entire Kingdom and also to make the public aware of the types of attacks that are targeting KSA. These reports provide information about the attacks that may be perpetrated against the Kingdom. We are continuously collaborating with CERT-SA for the purpose of making good use of the reports generated in the SAHNET lab at KFUPM. CERT-SA will also have the authority to communicate with other worldwide organizations and ISPs to notify them about any illegitimate activities that are initiated from their sites and targeting KSA. CERT-SA also represents the interface to other KSA government organizations and the corporate sector; therefore we believe that through the outcomes of the Saudi Honeynet Project, we have contributed to the improvement of cyber-security in the whole KSA.

The project has also institutionalized a sound framework for deploying Honeynets at the level of KSA for the purpose of securing and protecting KSA networks and Internet. Our Honeynet architecture and working model deployed at KFUPM can be extended to a larger and more complex Honeynet to be deployed at different locations in the Kingdom's networks. In fact intensive and productive discussions with CERT-SA about this matter are already underway.

Our researchers have tremendously benefited from the collaboration with top leaders in the field of network security and have participated in transferring such technology to KSA. In addition, the project has led to building local expertise, capacity, and knowledge base in installing, integrating, and developing Honeynets in KSA by training many students on such technology. Such expertise allows our team to provide consultancy for other local government and private organizations for the research and development as well as deployment of Honeynets in KSA. Some of the knowledge and experience gained has also been disseminated and shared with the GCC countries, including UAE CERT. The experience gained by the researchers can also be readily made available to other Saudi universities, private companies, and any other interested institution through consultations, lecture series, short courses, workshops, and/or prototype demonstrations.

As part of the project, a Honeynet lab has been established. The lab is being used extensively by undergraduate and graduate students for their research work, projects, and courses.

More importantly, different types of end users include KACST, CITC, CERT-SA, and other governmental institutions and local private companies can enormously benefit from the research and outcome of this project. For example,

1. KACST, the Internet provider for the whole Kingdom, can also benefit considerably from the results of this project. The outcome of this project can be used to increase the security of the Internet for the whole Kingdom and minimize information loss.
2. In general, major operators, providers, and regulators, including local ones such as KACST and Communication and Information Technology Commission (CITC) may use the findings of this project to better protect and secure the Internet for the whole Kingdom from cyber-attacks.
3. CERT-SA is a national organization that acts as a coordination center readily available to respond and tackle any emergency computer and network security incidents. As was stated above, CERT-SA is already making use of our findings resulting from this research project.
4. Kingdom wise, one of the major beneficiaries of the outcome of this project can potentially include many Saudi companies as well as governmental and non-governmental institutions that: (a) participate in deploying Honeypots and be partners in the Honeypot farm, or (b) participate in deploying “virtual Honeypots” that redirect traffic to the Honeypot farm.

10.0 BROADER IMPACTS OF THE STUDY

Teaching and Training

Faculty members from the team taught *Computer and Network Security* (1st semester of academic year 2009-2010 and 2nd semester of academic year 2010-2011), *Computer Network Design* (2nd semester of academic year 2009-2010), *Network Management* (1st semester of academic year 2010-2011), *Client Server Programming* (1st semester of academic year 2010-2011), and *Independent Research* (2nd semester of academic year 2010-2011) courses at the graduate level where the course project was directly related to the Honeynet Project. The course project involved literature review and/or simulations, prototyping, deployment, and lab experimentations of some of the proposed designs and solutions. In addition, *Computer and Network Security* (2nd semester of academic year 2009-2010), *Network Security Engineering* (2nd semester of academic year 2009-2010), and *Senior Design Project* (2nd semester of academic year 2010-2011) undergraduate courses were taught as well, and in which Honeynet development projects have been assigned to students.

In addition, other students have participated in the different parts of this project and were trained about Honeynets at different levels, including 2 PhD students, 6 M.S. students, and 8 undergraduate students.

As for the training part, two Malaysian Cybersecurity experts visited KFUPM as part of this project during the period of June 4-8, 2011. The program of their visit included delivering seminars on topics related to Android malware, malware evolution, and PDF attacks. In addition, two full days training was delivered on web security including hands on sessions and a mini cyber drill. A one day workshop, attended by invitation only, on analyzing malicious PDF was also held, which included a walk through on how to analyze in-the-wild malicious PDF files. The detailed program of the CyberSecurity Malaysia Experts Visit can be found in Appendix 4.

Infrastructure

The purchased equipment was used to setup a Honeynet lab which is being used by many students for research, projects, and courses. Malicious activities data is being collected continuously in the lab and shared with CERT-SA. This equipment will also help in conducting additional research and lab experimentations in different fields of networking. Similar equipment is available in the university but only in teaching labs that are dedicated for regular teaching activities and that cannot be shared with research projects.

Collaborations

The main collaboration was with CERT-SA, the Malaysian CyberSecurity, The Taiwan Honeynet Project Chapter, and the worldwide Honeynet Project. The collaboration with CERT-SA has been initiated in March 2011 and continued until now with exchange of ideas and the sending of reports generated in our lab to CERT-SA.

The collaboration with the Malaysian CyberSecurity, to which one of our consultants belongs, led to two experts visit in June 2011. This visit included consultancy on our Honeynet design and deployment as well as trainings and workshops to the larger KFUPM community (see Appendix 4 for more details).

This collaboration can be extended to KACST, Internet providers, regulators, and operators such as CITC, STC, Mobily; to help setting up Honeynets across the Kingdom and get useful feedback on the proposed nationwide Honeynet architecture.

Funding

The results of the project can lead to further research, and additional funding can be sought from NSTIP and/or internal KFUPM funding in the near future. The focus of the additional research may include areas such as Honeynet traffic analysis, malware analysis, and reverse engineering.

Contributions to the Strategic Technologies goals of NSTIP

Please see the appropriate table provided in section 7.

Others

Our ultimate goal is to deploy multiple Honeypots across the kingdom with the help of CERT-SA and to provide more awareness of the usefulness of having such solutions implemented as an additional security measure to the existing security infrastructure such as firewalls, IDS, and IPS. The setup of this infrastructure should also lead to more exchange of information related to the cybersecurity threats that the different organizations are potential target to; with the purpose of providing better protection. We hope through this that the project will have a wider impact and major benefits to the whole society.

11.0 OTHER CONCERNS

There are no additional concerns or comments related to the final reporting of the research program.

12.0 REFERENCES

[1] I. W. Stats. (2011, March 31, 2011). *Middle East Internet Usage and Population Statistics*. Available: http://www.internetworldstats.com/stats5.htm

[2] e.-G. P. Yesser. (2011, November 24, 2011). *e-Government Program – Yesser, Kingdom of Saudi Arabia*. Available: http://www.yesser.gov.sa/

[3] H. F. Lipson, "Tracking and Tracing Cyber-Attacks: Technical Challenges and Global Policy Issues.," CERT® Coordination CenterNovember 2002 2002.

[4] G. Keizer. (2005, Dutch Botnet Suspects Ran 1.5 Million Machines. *TechWeb News*. Available: http://www.techexchange.com/library/RFID%20And%20Dutch%20BotNets%20-%20September%202005.pdf

[5] J. Leyden. (2004, Telenor takes down 'massive' botnet - Clients are still zombies. *Enterprise Security - The Register*. Available: http://www.theregister.co.uk/2004/09/09/telenor_botnet_dismantled/

[6] J. Leyden. (2005, ISPs urged to throttle spam zombies - International clean-up campaign. *Spam - The Register*. Available: http://www.theregister.co.uk/2005/05/24/operation_spam_zombie/

[7] T. Weber. (2007, Criminals 'may overwhelm the web'. *BBC News*. Available: http://news.bbc.co.uk/2/hi/business/6298641.stm

[8] R. L. J. Levine, H. Owen, D. Contis, and B. Culver, "The use of Honeynets to detect exploited systems across large enterprise networks," in *Information Assurance Workshop, 2003. IEEE Systems, Man and Cybernetics Society*, United States Military Academy, West Point, NY, 2003, pp. 92-99.

[9] L. Spitzner, *Honeypots: Tracking Hackers, Addison-Wesley, http://www.tracking-hackers.com/book/*, 2003.

[10] G. P. Spathoulas and S. K. Katsikas, "Reducing false positives in intrusion detection systems," *Computers & Security*, vol. 29, pp. 35-44, 2010.

[11] S. Corp. (2003, Symantec Releases Decoy-Based Intrusion Detection System. Available: http://www.symantec.com/press/2003/n030623b.html

[12] Honeynet, "Know Your Enemy: GenII Honeynets," 2005.

[13] Securitydocs. Honeypots Revealed. Available: http://www.securitydocs.com/library/2692

[14] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Computer Survey*, vol. 41, pp. 1-58, 2009.

[15] A. Dainotti, A. Pescape, and G. Ventre, "NIS04-1: Wavelet-based Detection of DoS Attacks," in *Global Telecommunications Conference, 2006. GLOBECOM '06. IEEE*, 2006, pp. 1-6.

[16] J. Haggerty, T. Berry, Q. Shi, and M. Merabti, "DiDDeM: a system for early detection of TCP SYN flood attacks," in *Global Telecommunications Conference, 2004. GLOBECOM '04. IEEE*, 2004, pp. 2037-2042 Vol.4.

[17] D. Ping and S. Abe, "Detecting DoS attacks using packet size distribution," in *Bio-Inspired Models of Network, Information and Computing Systems, 2007. Bionetics 2007. 2nd*, 2007, pp. 93-96.

[18] P. Barford, J. Kline, D. Plonka, and A. Ron, "A signal analysis of network traffic anomalies," presented at the Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurment, Marseille, France, 2002.

[19] A. Lakhina, M. Crovella, and C. Diot, "Mining anomalies using traffic feature distributions," presented at the Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications, Philadelphia, Pennsylvania, USA, 2005.

[20] G. Nychis, V. Sekar, D. G. Andersen, H. Kim, and H. Zhang, "An empirical evaluation of entropy-based traffic anomaly detection," presented at the Proceedings of the 8th ACM SIGCOMM conference on Internet measurement, Vouliagmeni, Greece, 2008.

[21] F. Al-Haidari, M. Sqalli, K. Salah, and J. Hamodi, "An entropy-based countermeasure against intelligent dos attacks targeting firewalls," presented at the Proceedings of the 10th IEEE international conference on Policies for distributed systems and networks, London, United Kingdom, 2009.

[22] *Information Entropy, [Online]* *http://www.absoluteastronomy.com/topics/Information_entropy*.

[23] Honeynet.org. *Honeynet Project, Honeynet Definitions, Requirements, and Standards Documentation, Honeynet Project website (http://old.Honeynet.org/alliance/requirements.html)*.

[24] D. Watson and J. Riden, "The Honeynet Project: Data Collection Tools, Infrastructure, Archives and Analysis," in *Information Security Threats Data Collection and Sharing, 2008. WISTDCS '08. WOMBAT Workshop on*, 2008, pp. 24-30.

[25] E. Balas and C. Viecco, "Towards a third generation data capture architecture for Honeynets," presented at the Information Assurance Workshop, 2005. IAW '05. Proceedings from the Sixth Annual IEEE SMC
2005.

[26] D. F. Gong, "Deciphering Detection Techniques: Part II Anomaly-Based Intrusion Detection " 2003.

[27] A. Kind, M. P. Stoecklin, and X. Dimitropoulos, "Histogram-based traffic anomaly detection," *Network and Service Management, IEEE Transactions*, vol. 6, pp. 110-121, 2009.

[28] O. Thonnard and M. Dacier, "A framework for attack patterns' discovery in Honeynet data," presented at the Digital Investigation, 2008.

[29] P. Barford, Y. Chen, A. Goyal, Z. Li, V. Paxson, and V. Yegneswaran, "Employing Honeynets For Network Situational Awareness," in *Cyber Situational Awareness*. vol. 46, S. Jajodia, P. Liu, V. Swarup, and C. Wang, Eds., ed: Springer US, 2010, pp. 71-102.

[30] S. I. project, "SURFcert IDS Development homepage."

[31] XMPP Server. Available: http://carnivore.it/2010/10/13/xmpp_server

[32] carniwwwhore. Available: http://carnivore.it/2010/11/27/carniwwwhore

[33] Honeynet.org. *http://www.Honeynet.org/challenges*.

[34] hack.lu, "Information Security Visualization Contest, hack.lu 2009, http://2009.hack.lu/index.php/InfoVisContest," 2009.

[35] http://www.wireshark.org/. *wireshark*.

[36] http://www.netresec.com/?page=NetworkMiner. *Network Miner*.

[37] W. Lu and A. A. Ghorbani, "Network anomaly detection based on wavelet analysis," *EURASIP Journal on Advances in Signal Processing*, vol. 2009, pp. 1-16, 2009.

[38] Metasploit. *Metsploit Framework - Metasploit Project web page*, *http://www.metasploit.com*.

[39] Fyodor.:. (2007, Top 100 Network Security Tools (last visited, July 25, 2007), available on line on http://sectools.org.

[40] P. Laskov and M. Kloft, "A framework for quantitative security analysis of machine learning," presented at the Proceedings of the 2nd ACM workshop on Security and artificial intelligence, Chicago, Illinois, USA, 2009.

[41] K. Rieck and P. Laskov, "Language models for detection of unknown attacks in network traffic," *Journal in Computer Virology*, vol. 2, pp. 243-256, 2007.

[42] P. Düssel, C. Gehl, P. Laskov, J.-U. Bußer, C. Störmann, and J. Kästner, "Cyber-Critical Infrastructure Protection Using Real-Time Payload-Based Anomaly Detection," in *Critical*

Information Infrastructures Security. vol. 6027, E. Rome and R. Bloomfield, Eds., ed: Springer Berlin / Heidelberg, 2010, pp. 85-97.

[43] D. L. Olson and D. Delen, *Advanced Data Mining Techniques*, 1st ed.: Springer Publishing Company, Incorporated, 2008.

13.0 APPENDICES

13.1 Appendix 1

Trip Report for the Saudi Honeynet Project Team to the National Computer Emergency Response Team, Malaysia

Dr. Zubair Baig

As part of the KACST National Science Technology and Innovation Plan's Saudi Honeynet Project, a team comprising of two project members, namely, Drs. Zubair A. Baig and Marwan Abu-Amara, paid a visit to the CyberSecurity headquarters, Malaysia. The purpose of the visit was to foster stronger presence of the Saudi Honeynet chapter in the global Honeynet community, and to establish links with the prime department for addressing Internet threats in Malaysia, i.e. CyberSecurity Malaysia. Malaysia Computer Emergency Response Team (MyCERT) is a part of CyberSecurity Malaysia, and reports directly to the Ministry of Science, Technology and Innovation. In addition to the MyCERT, CyberSecurity Malaysia is also constituted of the National999 facility for receiving emergency calls associated with online crime, round the clock. The third wing of the organization is the Malware Research Laboratory, wherein, constant monitoring and analysis of network data is performed, to detect known and existing threats against the cyberinfrastructure of Malaysia. In addition, work is being carried out to study the possibility of establishing an early warning system, to detect unknown types of online attacks.

Upon arrival at the MyCERT premises, the two member project team of the Saudi Honeynet project, was hosted by Mr. Adli Abdul-Wahid, Director, MyCERT, and his staff. Several discussions unfolded during the course of the day, appertaining to mutual cooperation and sharing of technical know-how for helping achieve our project objectives. MyCERT also provides constant and up-to-date advisories on current and potential threats against software that is engaged on the Internet. For instance, a flaw in a particular web browser, if successfully diagnosed by the team, will be posted on to their website, as an advisory to all computer users across the country. The Malaysian Honeynet project is also known as Lebahnet, which began around 7 years back as primarily a data collection and basic analysis project. At present, the project has evolved to the next level of zero-day attack detection and high quality visualization of threats.

It was also learnt during the visit that Distributed Denial of Service Attack drills are regularly performed by MyCERT, to test the capability of an organization to withstand such an attack, if at all it takes place. Such drills provide corporations with the essential know-how, to facilitate correcting their network and system configurations, and installing the necessary patches in time before catastrophe befalls.

The MyCERT is also actively involved in rapid deployment of hardware Honeynet sensors, which can be purchased for nearly USD 1400, within corporate premises of local companies, with the intent of providing MyCERT with access to critical datasets, for analysis and study. MyCERT is also an active provider of training to analysts in diverse areas such as Incident handling, Network Security, Forensics, and Penetration Testing.

Issues raised and addressed with regards to the Saudi Honeynet setup and configuration:

1. It is suggested to avoid the use of honeywalls as such, as they are outdated and not updated by active members. Instead, it was recommended by MyCERT to avoid the data controller use in its entirety for the project, but rather focus on the use of emulator tools such as Sguil, which is a TCL-based network security monitoring and analysis tool. Such tools will work well with low interaction Honeypots. In addition, the use of network flowtracker will prove to be a safe solution for Honeypots, as it does not cause problems if compromised.
2. The purpose of Honeypot deployment in corporations is primarily to detect malwares that may be causing damage to the corporate network and systems. No such information is revealed if a high-interaction Honeypot is used. It is therefore essential to use low-interaction Honeypots.
3. Only open certain important ports when a Honeypot is deployed in a corporation. For instance, ports 445, 8080, 8081.
4. Use HTTPS for secure transfer of captured network traffic from a Honeypot (sensor) back to the central repository, for further analysis.
5. Wireless networks are no different from wireline networks, and the threats to both can be considered to be the same, albeit with the addition of a rogue Access Point threat, which is unique to wireless networks.
6. For purposes of network traffic flow tracking, it is suggested to use tools such as FlowD, NFCapd, and NSFlow. For observing statistical data such as the length of traffic observed, NetFlow is an effective tool.
7. The purpose of full packet capture is solely for verification purposes. Otherwise, large scale redundant data will have to be scanned through frequently, without any results.
8. For the KFUPM Honeynet deployment, it was suggested by the MyCERT team to have honeywall operate as a controller and not as a traffic collector. Other tools, as mentioned above, may be used for the latter purpose.

9. For the KFUPM distributed Honeynet, it was suggested to perform subnetting for each type of network (ITC, ADSL, etc.), and isolate each specific Honeypot from the rest of the network. Otherwise, a ping-pong match will take place between the different Honeypots operating in the networks.
10. For study and analysis of worms with known signatures, low interaction Honeypots are very effective.
11. High interaction Honeypots are more of use for the study and extrapolation of zero-day attacks (anomalies from known anomalies), and where not much knowledge regarding an attack exists. However, such types of Honeypots will generate a lot of noise, which will be hard to maintain.
12. Several in-house tools for data analysis are being developed by MyCERT. Some of them include: pKAJI for analysis, Gallus for Pdf Analysis, Mykotakpasire data analyzer etc.
13. Some of the suggested topics for further research were: Automated data analysis, Malware analysis, Honeynets for IPv6/VoIP/3G/4G networks.
14. The team offered the Saudi Honeynet chapter with free data set samples, for possible analysis by the group.
15. The team strongly suggested that we establish links with Saudi CERT, as our research findings will be critical to correct and smooth malware-free operation of the Kingdom’s IT infrastructure.
16. An antiphising conference is being organized by MyCERT for April, to be held in Kuala Lumpur
17. The global Honeynet meeting for next year will be held in Paris, late February.

13.2. Appendix 2

Trip Report of the Visit to Saudi Community Emergency Response Teams (CERT-SA) Riyadh on Wednesday March 16th, 2011

Dr. Mohammed Houssaini Sqalli
Principal Investigator
Saudi Honeynet Project
NSTP Project # 08-INF101-4

April 20th, 2011

As part of the KACST National Science Technology and Innovation Plan's Saudi Honeynet (SAHNET) Project, a team comprising of two project members, namely, Dr. Mohammed H. Sqalli and Dr. Talal M. AlKharobi, paid a visit to the Saudi Community Emergency Response Teams (CERT-SA) in Riyadh on Wednesday March 16th, 2011.

As for the meeting with the team from the Prince Muqrin Chair (PMC) for IT Security which was planned during the same week, it has been cancelled as it was not possible for the PMC team to meet with us.

The purpose of the visit was to meet with the CERT-SA team to discuss ways to collaborate on issues related to network security, discuss ways to deploy Honeynets at the KSA level, and the initiate three-way collaboration between our team (SAHNET), CERT-SA, and Malaysian CERT (MyCERT).

Upon arrival at the CERT-SA premises, the two member SAHNET project team was hosted by Mr. Mohammad S. AlArifi, Information Security Specialist, CERT-SA, and his team. Several discussions unfolded during the course of the day, appertaining to mutual cooperation and sharing of technical know-how for helping achieve our project objectives. CERT-SA also provides constant and up-to-date advisories to government agencies within the Kingdom based on reports it receives from several organizations such as Shadowserver concerning the current status of the attacks being initiated from within Saudi Arabia and targeting the Internet at large.

The main activities that were held during the visit are the following:

1. Dr. Mohammed H. Sqalli gave a presentation to the CERT-SA team about the work that has been completed so far as part of the SAHNET project. The CERT-SA team appreciated the work achieved and the results obtained so far by the SAHNET project. For instance, they were impressed by the information presented with respect to the detection and analysis of many rogue activities that were seen on our deployed Honeynet at KFUPM.
2. We have also proposed a framework for the deployment of Honeynets across the Kingdom based on the current KFUPM Honeynet deployment. This requires further discussion with the CERT-SA team after we finalize our prototype within KFUPM.
3. We visited the different departments within CERT-SA and learned more about the activities of CERT-SA, mainly the notifications they send to government agencies based on the reports they receive from international security organizations such as Shadowserver. We also discussed and provided feedback on an application they are developing for the automation of part of this process to make the notifications more efficient.
4. We have also discussed some of the issues with respect to the Security Operation Center (SOC) which KFUPM (through ITC) is going to be part. We also got some answers on questions that ITC security team at KFUPM had about the SOC project.
5. We have briefly discussed the relationship of both of our teams with the Malaysian CERT (MyCERT).

We believe that the trip was very successful and that the collaboration between CERT-SA and SAHNET at KFUPM will allow for providing better security at the level of the Kingdom. We also believe that the outcome set for the visit and which is to collaborate and propose ways to improve security at the KSA level, and more specifically for academic institutions has been achieved. This is based on our proposed framework for a Kingdom wide deployment of Honeynets, and which could begin by initially involving Universities within the Kingdom.

13.3 Appendix 3

Trip Report of the Visit to the

Honeynet Project Annual Workshop

ESIEA Institute, Paris, France
March 21st-25th, 2011

Dr. Mohammed Houssaini Sqalli
Principal Investigator
Saudi Honeynet Project
NSTP Project # 08-INF101-4

As part of the KACST National Science Technology and Innovation Plan's Saudi Honeynet (SAHNET) Project, a team comprising of two project members, namely, Dr. Mohammed H. Sqalli and Dr. Farag Azzedin, participated in the Honeynet Project annual workshop that was held in the ESIEA Institute, Paris, France during March 21st-25th, 2011.

The purpose of the trip was to attend the Honeynet Project annual workshop, and meet with members of various national Computer Emergency Response Teams (CERT), experts from leading technologies companies, and professors from various universities. And, the aim of attending this event was to learn and discuss security issues related to our NTISP funded projects.

This workshop was an excellent opportunity to discuss state-of-the-art security and trust issues with professionals including the security expert, Lance Spitzner (Honeynet Project founder and former CEO), Christian Seifert (Honeynet Project CEO), David Watson (Honeynet Project Chief Research Officer), and many members of the Board Of Directors and officers of the Honeynet Project from around the world. In addition, many experts from leading technologies companies as well as professors from universities attended the workshop. Among the experts that were invited, we met Rogier Spoor from SURFnet in Netherlands and Piotr Kijewski from CERT Polska. We have established connections with most of these experts for future collaborations.

This workshop was also an opportunity to meet many members of other Honeynet chapters from different countries such Malaysia, USA, Germany, Italy, and others. We have also discussed some specific issues with respect to the design and deployment of the Saudi

Honeynet project as well as the Analyzer tool that we have built as part of a student's MS thesis work. We have learned many new things that will help us improve our NTISP funded project deliverables, including the latest trends and research focus of the Honeynet Project community.

We also believe that such visit boosted our experience and awareness of the high priority issues in terms of network security worldwide, and more specifically Honeynet deployment.

The main activities that were held during the visit are the following:

1. We attended the Public day where many presentations were delivered and which helped us get a better understanding of some of the work being carried by different chapters of the Honeynet Project. All content of the public 2011 Honeynet Project workshop have been uploaded to the Honeynet Project website. The presentation abstracts, slides, and most of the videos can be accessed from: https://www.Honeynet.org/SecurityWorkshops/2011_Paris
2. We attended the closed sessions. Information on these days should not be disclosed to the public. But, it was much more useful for us as it covered more interesting and important topics related to the Honeynet Project.
3. We participated in the hands on workshops on reverse engineering of PDF and Android malwares. We were also provided with virtual machines that include all the necessary material of perform the reverse engineering, including files that have malwares in them.
4. During the closed session, Dr. Mohammed H. Sqalli gave a presentation on the topic "Honeynet Traffic Analyzer Using Anomaly Detection Techniques" where he presented the work that has been accomplished as part of Mr. Syed Naeem Firdous' MS thesis work. The work is about developing a novel technique that combines feature-based and volume-based parameters to analyze Honeynet traffic. The proposed technique has been used on traces provided by the Honeynet Project community as well as traces that were collected at KFUPM. The technique has shown very good results that will help the Honeynet Project community in the analysis of the traffic collected on the Honeypots deployed. We also received very constructive feedback from the community about this work. In addition, we plan to make a tool available to the community which is based on the developed technique.
5. During the closed session, Dr. Mohammed H. Sqalli presented a problem that was seen on our KFUPM network about SQL to get feedback from the community. Nobody from the attendees has seen a similar problem. Therefore, it was agreed to make the trace available to the community, which was done, so that other can investigate it more.

6. We had meetings with many experts. For instance, we have discussed with few experts, and more specifically Rogier Spoor and Piotr Kijewski ways on how to deploy Honeynets at a larger scale at the level of the Kingdom where we also plan to benefit from our newly established connection and collaboration with the Saudi CERT. We plan to propose a Framework for a larger Honeynet deployment at the level of the Kingdom.
7. We obtained more information from Mark Schloesser about the steps needed to decode some code that we captured on our Dionaea Honeynet, and which our team was unable to decode. This is related to the malware download.
8. We have agreed with other teams to exchange traces and other information, and which we are following up on. For instance, we have already received more than 20GB of traces.
9. We have started some collaboration with other teams on some research topics. For instance, we initiated some discussion with Max Kilger and Thomas J. Holt on their work related to the civilian Cyberwarrior study. The study is about exploring individual attitudes toward and support for acts of political unrest in virtual and real environments.
10. We have learned about the latest tools being developed and used by the community such as: Dionaea, Glastopf, hpfeeds, streams, and others.
11. We got some of the tools installed and working on our laptop with the help of the tool's developer, such as: Glastopf and streams.

We feel that the trip was very successful; including initiating the collaboration with other Honeynet Project chapters. We have already started many positive interactions with the Honeynet Project community and which lead to many useful activities, including:

1. We received a large set of traces from few Honeynet Project members worldwide.
2. We were assigned the task to translate to Arabic the SANS newsletter, namely OUCH! which is widely read around the world.
3. We were invited to participate in a survey on social engineering related to the study on civilian Cyberwarrior.

Few other points that we have learned in the workshop include:

1. The HP recommends the use of open source whenever possible so that the work can be made available and be used by the community. This may have some implications on the framework our team was working on. For instance, the use of Oracle is one issue that we need to look at.

2. It was stated by David Watson, the HP Chief Research Officer, that Dionaea is considered by the community as the preferred tool. Therefore, we decided to continue using it for our Honeynet deployment.
3. The blackhat community is getting very sophisticated and dynamically organizing itself to evade existing defenses, which opens the door for a large scale cyberwar. Lance James proposed a longer term perspective of defense by utilizing carefully crafted offense both legally and digitally to gain an upper hand.

Finally, and as future directions for the Honeynet Project, the following are the six R&D focus areas of the Honeynet Project in 2011, according to the Honeynet Project Chief Research Officer:

1. Mobile device Honeypots
2. Virtualization Honeypots and attacks (e.g., Hyperviser)
3. Topical malware (stuxnet SCADA, etc.)
4. Active defense research (e.g., take botnets down in an ethical manner)
5. IPv6 Honeynets
6. Distributed data collection, analysis, and visualization (including HonEeeBox)

In summary, we believe that the outcome set for this visit and which is to plan for enhancing our actual KFUPM Honeynet design and the proposed Honeynet for Saudi Arabia has been achieved. In addition, this visit allowed us to initiate collaboration with many Honeynet chapters and we are still benefiting from this.

13.4 Appendix 4

Program at KFUPM of the

CyberSecurity Malaysia Experts Visit

June 4-8, 2011

Organized by the Saudi Honeynet (SAHNET) Chapter
Sponsored by KACST under the 1st five years NSTIP
Hosted by the Computer Engineering Department

Saturday afternoon

- Meeting with KFUPM IT Center security team
- Saudi Honeynet (SAHNET) Project review and discussion
- Meeting with KFUPM Faculty members

Sunday, 10:00AM-1:00PM

- **Half day seminar, including the following topics:**
  - The rise of Android malware
  - The malware evolution
  - Targeted attack: Are you ready for PDF Attacks?

Sunday afternoon

- Meeting with KFUPM Faculty members & Students
- Lab visit

Monday and Tuesday, 8:30AM-3:30PM

Web Security Training:

- Web application security is critically important - today, over 75% of hacker attacks worldwide are actively targeting Web applications. In this rapidly evolving landscape, professionals -- developers, IT, management, and information security -- have an important part to play in Web application security. Our courses provide the up-to-date knowledge and skills required to understand and deliver meaningful security measures.
  - Hands on session. Training materials will be provided
  - Mini cyber drill on the second day, so participant can have real experience to defend their server using knowledge gained on day 1 of the training.
  - Attendees may use their own laptops or one of the PCs available in the lab. They need to have Administrator privileges and VM workstation installed on their laptops.
  - Maximum 20 person (First Come First Serve)

Wednesday, 8:30AM-3:30PM

Analyzing Malicious PDF Workshop:

- This workshop will walk through participant how to analyze in-the-wild malicious PDF. We'll share how we can analyze malicious PDF file by using publicly available tools. Shellcode analysis will be conducted as well to get the whole picture of PDF Attack Anatomy. The analysis of malicious PDF will start with a simple innocent PDF to highly obfuscated PDF file. A few malicious samples applied obfuscation techniques such as PDF syntax obfuscation, JS obfuscation, will be analyzed. For the version 2, we'll analyze more advance samples, which implemented many obfuscation techniques to make analysis more difficult.

- We expect participants to have basic knowledge on PDF document structure, exploit structure, as well as shellcode. We'll provide VM training image for the training

    - Hands on session. Training materials will be provided.
    - Audience is required to have access to administrator privilege to install application.
    - Maximum 20 person (First Come First Serve)

Biography of Mr. Mahmud Ab Rahman:

Mahmud Ab Rahman currently works as Information Security Specialist for Malaysia Computer Emergency and Response Team (MyCERT) under umbrella of CyberSecurity Malaysia. Prior to that, he worked as an Intrusion Analyst at MyCERT department. His education background comprises of Master Degree in Computer Science from National University of Malaysia in 2006. Prior to that, he obtained a Degree in Computer Science from the same university.

Mahmud has been involved in the computer security field for over 6 years. His area of focus and interest is network security, Honeynet, botnet monitoring, and malware analysis. He also engages in several large scale penetration-testing exercises and to provide solutions for any vulnerability detected. Moreover, he is recognized for conducting numbers of training for organizations to talk on topics ranging from introduction to advanced security courses. He is a occasional speaker at conferences such as FIRST Conference,Honeynet Annual Workshop, FIRST-TC, CSM-ACE and Infosec.MY. He is currently certified for SANS's GPEN (gold) and GREM.

Biography of Mr. Mohd Hafiz Bin Mat Tabrani:

Mohd Hafiz Tabrani currently works as Senior Intrusion Analysis for Malaysia Computer Emergency and Response Team (MyCERT) under umbrella of CyberSecurity Malaysia. Prior to that, he worked as an Intrusion Analyst at MyCERT department. His education background comprises of Degree in Computer Science from National University of Malaysia in 2000.

Hafiz has been involved in the computer security field for over 5 years. His area of focus and interest is network security, Honeynet, websecurity and malware analysis. He also engages in several penetration-testing exercises and to provide solutions for any vulnerability detected. Moreover, he is recognized for conducting numbers of training for organizations to talk on topics ranging from introduction to advanced security courses. He also involved as a GSOC (Google Summer of Code) mentor for Honeynet Project during 2010 mentoring on PHP Sandbox. He is also main contributor for CyberSecurity Malaysia Honeynet Project's blog. He currently holds a GPEN certification from SANS Institute.

13.5 Appendix 5

Report about the Visit to Taiwan Honeynet Chapter

The visit to the Taiwan Honeynet chapter was scheduled from the 27th of June to the 1st of July 2011. The visit was initiated upon a request made by the Saudi Honeynet chapter project (SAHNET) to the Taiwan Honeynet project chapter. The later was very keen to accept the request and welcome the SAHNET members, i.e., Dr. Talal Al-Kharobi and Mr. Hakim ADICHE, both from COE department, and scheduled and organized the visit.

The purpose was to learn and benefit from the experience gained by the Taiwan Honeynet chapter project, and to have an insight into the various technological developments, improvements, trends, and results that were achieved and centered on the Taiwan Honeynet chapter project.

Visit to the Tainan Science Park

The National Center for High Performance Computing (NCHC) is located in the Science Park, in Tainan County, about 10 kilometers from Tainan city. This center is certified ISO 27001:2005 for security info: Management Certification. Its role is to provide service, research, as well as training in multidisciplinary domains related directly or indirectly to the IT technology. Among services provided by the NCHC are the High Performance Core - HPC services. They are aimed at providing storage service, training service, high performance computing, and networking.

The NCHC is part of the Taiwan Research and Education Network (TWAREN), a completely independent network from the Internet but connected to. It relies on technologies such as STM-64, STM-16, and 10-GE for connecting the different backbone nodes. TWAREN connects the top 10 universities and research centers in Taiwan. It is also connected to a similar network in the USA.

In the NCHC, open source software tools are developed. Some of these tools related to our Honeynet project are the DRBL (Diskless Remote Boot in Linux), and Clonezilla which is a software used to restore backups and for disaster recovery. The NCHC network is used for inter-disciplinary science and technology research and provides a support for educational activities, as well.

In the part related directly to the Honeynet project, and among other roles, NCHC is responsible for malware and abnormal traffic analysis.
It has strong ties with CERT-Taiwan and performs digital forensic as well as reverse engineering of Malware code. The Malware reverse engineering is just starting and not yet a fully deployed activity. The botnet detection and behavior analysis are other activities performed in the NCHC center. One interesting thing is that they use Rainbow table generator for reversing cryptographic hash functions and hash cracks based on cloud computing and high performance computing. They rely mainly on information mining technology in their analysis tasks.

Their Honeynet network components deployed so far can be described briefly as follows:

1. The monitoring platform is based on Intel blade.
2. The Honeypots run on top a virtual environment such as VMware sphere.
3. The information search engine is implemented using Splunk software.
4. The collection, display and arrangement of information are achieved with a security dashboard developed by the NCHC center.

It should be noted that Splunk is a commercial product and that a free copy of it can analyze a file having 5 MB of size as maximum. The file is fed manually to Splunk for analysis.

There are 8 universities with Honeynets in Taiwan. Some Honeypots use Nepentheses while others run Dionaea. However, Nepentheses is being phased out gradually to be replaced by Dionaea.

The IDS used is SNORT; an open source intrusion detection system software. They also use Surf-IDS for logging malicious traffic traces. However, hardware IDS appliances are preferred for logging malicious activities on high speed links. The data is collected and stored using a Storage Area Network (SAN) facility with high storage capacity. All reporting of alerts, alarms, and critical thresholds detections are based on SNMP protocol and Syslog utility.

NCHC relies on Arc-Sight and IPS for logging malware activities traces. The Arc-Sight is commercial software that analyses data in real time and provides log activities. Its main drawback is that it is very expensive; around 100K USD per year for license renewal only. The operating system used so far for the Honeypots deployment is **Guest OS**. Malware attacks are detected using IDS. The IDS thresholds are tuned to achieve best results.

An infected machine will have its image re-installed and IP address changed. At this stage, only VMware machines are used for running Honeypots. Every infected machine will have

the IP address of the bot-master machine reported to the system administrator of the network from where the attack is detected. The IP address is tracked from the log files.
Geo IP localization is used for locating the bot-master machines or C&C bots from where other bots are commanded and controlled. *Their research focuses on ssdeep* **info** for new malware detection using fuzzy ontology and FML-based ontological layout. Basically, they define the following variables before fuzzifying them; File Hash, Connect IP, File Changes, and Similarity.

In order to find out the values for the above variables and check if a malware has infected a machine, the following algorithm has been devised:

1. Create a clean image
2. Setup Auto-run
3. Create/restore client image
4. Set forensic baseline
5. Download and run malware
6. Dump memory image and capture network traffic
7. Reboot to Linux and save infected client image
8. Mount infected client image
9. Do analysis
10. Restore clean image to client – go back to step 5

To clone images, they use Clonezilla software developed locally as open source software. To compare between the clean and infected image, AIDE (Advanced Intrusion Detection Environment) software tool is used. The malware analysis is restricted to real Operating Systems, instead of virtual machines. This is because some Malware have tendency to detect a virtual environment and turn off their activities. With a real Operating System, more Malwares are detected, according to their results.

They use InetSim and Sandnet server as a Sandbox to simulate Internet services and attract potential attackers, in replacement to Truman Sandbox.

As a summary of some of the tools used:

1. Log collection and analysis: Arc-Sight and Splunk
2. Monitoring: ngios
3. Performance evaluation: cacti
4. Ticket system: remedy
5. Integration: dashboard

Some of the common software tools that help localize IP addresses geographic location:

1. Rumint: classify IP traffic by country using GeoIP
2. Logstalgia: for web site traffic virtualization

Their communication with CERT, to whom they send suspected IP addresses for further investigation as a possible source for Bot-Master or Bot operations, is well established. The IP addresses are reported if the number of hits goes above a well defined threshold. The reporting of IP addresses is not acknowledged by CERT due to lack of manpower. However, CERT people will try to contact the administrators of the servers from where attacks are initiated to take further actions. They can only enforce the administrators of servers that belong to public sector to comply with their requests. The same cannot be done with servers that belong to ISPs and which are used by non-governmental sector. The relation between the different Honeynets and CERT is based on trust established through personal relation.

Visit to the NCKU Network and Data Center

A cloud computing network has been developed at the NCKU. The cloud is based on VMware vSphere, VMware Center, and VMware View. For the latter, the Citrix Xen Desktop is more preferred. Although the center is non-profit, some of its services are rented to provide high processing power to the research community such as faculty and researchers. The rent fees are very affordable for whoever needs more processing power instead of relying of personal PCs and servers.

One of the most important software used is Emulab. It is developed locally and serves as a management and control platform for laboratory activities and research experiments. Any user willing to use such platform can choose the type of network, the operating system, the node type, and application to install on a virtual environment and based on that, he can create a virtual networking environment for his experiment and research. Some of the techniques used in the center are based on Open-Flow and Net-FPGA. The analysis of Malware at this level relies on Malbed test-bed and a Malbed report is generated.

Visit to Taiwan Academic Network CERT – TACERT

TACERT is located in the Sun Yat-Sen University in Kaohsiung, south of Taiwan (www.tacert.edu.tw). TANET is a network connecting schools, and universities together, in Taiwan. This network is constantly monitored by TACERT for any malicious behavior and hacking activities. Due to E-government and E-commerce, TACERT is responsible for network security at the country level. This is very important especially that 25% of the attacks are from within the country.

An annual drill is organized by TACERT and aimed to train the network administrators of the schools and universities to take appropriate actions to secure and protect their networks whenever it is necessary. The alerts are received from the Security Operation Center – SOC. It is the responsibility of SOC to tune the alert parameters in order to catch any malicious activities. The tuning of these parameters is very important and affects the volume of alerts generated. The processing time for incident report has been reduced to few hours and TACERT can be contacted through Telephone, Email, Fax, and Forum dedicated to this purpose.

As part of research work related to the Honeynet program, one version of Honeypot software is being developed. This Honeypot software is named BASHPOT. It is considered as a high interaction Linux based Honeypot.
This Honeypot controls command usage and establish a white-list of command allowed to be executed on Linux platform and denies the rest of commands.

Visit to ACER

The visit to Acer was very interesting. We visited their center of security monitoring which allows receiving and reporting all alerts initiated by some events which indicate security problems. We were also briefed about different scenarios and techniques used to identify DOS attacks as well as Malware attacks. We later on visited their Backup center which was really an impressive one. This backup center provides solutions for data backup for different companies around the world.

Benefit to the SAHNET Project

Some of the benefits of this visit that can be applied to the SAHNET project are listed below:

1. The Taiwan Honeynet project network is implemented by experienced engineers and professionals and does not rely on students (undergraduate or graduate). The students are left with research and analysis tasks. The NCHC networking staff are all working on different projects and one of them is the Honeynet. This is to indicate that they do not have fully dedicated people for the Honeynet platform deployment. The same paradigm can be followed in our SAHNET project.
2. Develop, organize, and implement a Security Operation Center – SOC. The architecture should be achieved by experienced and professional teams but the analysis of data can be left to the students as part of their project and research.

3. Coordinate with CERT-SA for implementing an annual drill that will allow different organizations to test their network security and improve the responsiveness of their specialized staff.
4. Start using some of the Taiwan NCHC open source software such as DRBL and Clonezilla. Later on, similar open source software can be developed locally and integrated in the SAHNET.
5. Tune our SAHNET Honeynet platform using protocols, utilities, and environment for achieving better results in capturing Malware activities as well as abnormal networking behavior. Similar or parallel approach as the one followed by Taiwan Honeynet project can be adopted and later on modified.
6. Establish a trust relationship with CERT-SA members following the same format as the one used by TACERT and based on personal relationship. We can enforce some integrity check to Email exchanges between SAHNET-KFUPM and CERT-SA.
7. Explore open source programs such as ngios and cacti for monitoring and performance measure.
8. Find other alternatives to ARC-Sight and Splunk as open source software.
9. Improve the architecture of the Honeynet platform by integrating different solutions and protocols and including virtualization as well as real machines.
10. Build a platform for Malware analysis and reverse engineering.

Conclusion

The visit to the Taiwan Honeynet chapter project was very enlightening. We have seen how they integrate many networking solutions and protocols in order to deploy their Honeynet platforms. We also got exposed to their relation and ways of collaboration with TACERT (CERT-Taiwan) in handling security issues and mitigating networks attacks.

We mainly recommend the following:

1. Inviting the director of the cloud computing center at NCKU to give a presentation or seminar about their experience in cloud computing platform development and how they succeeded in establishing such an asset, and its impact on Taiwan research and economy. Also, the integration of cloud computing with the Honeynet project for malware detection and analysis.
2. Establish cooperation with them for Malware analysis and reverse engineering of Malware. They also requested to establish such cooperation with KFUPM.
3. Host the annual Honeynet chapter conference in KFUPM in collaboration with CERT-SA. The impact of organizing such a conference will have lot of benefits for

local universities, research centers, and industries related to network security and willing and interested in establishing their own Security Operation Centers.

Hakim ADICHE

Lecturer, COE department
King Fahd University of Petroleum and Minerals
Dhahran, Saudi Arabia
Email: adiche@kfupm.edu.sa
Mobile: +966591970340

13.6 Appendix 6

King Fahd University of Petroleum & Minerals
KACST's National Science, Technology, Innovation (NSTIP) Plan

The Saudi Honeynet Project
Summary Report on the 1st Pilot Run in a Real Network

For the Period of
November-December 2010

Project Lead: Dr. Mohammed H. Sqalli
Edited by: Dr. Zubair Baig

Team Members:
Syed Naeem Firdous
Muhammad Shoieb Arshad
Azzat Ahmed Al-Sadi

June 11th, 2011

1. Introduction

This summary report provides a summarized account of the 1st pilot run undertaken by the Saudi Honeynet project team in a real network deployment on KFUPM premises. The detailed report is available upon request. The Honeypot tool used as the standard platform to run the pilot experiment was *Dionaea*. The pilot deployment consisted of two phases. During the first phase (November 6th & 9th, 2010), we placed the Dionaea-based Honeypot on the public Internet within ITC premises, whereas in the second phase (November 17th, December 3rd, 2010), the Honeypot was placed on the ADSL network, which facilitates Internet services for the faculty housing area. During this activity, we collected network traffic and results therein; some of which are discussed below.

As shown in Figure 1, the location of the Dionaea Honeypot was on the Internet end of the ITC network, for the first pilot run. In this scenario, we placed the Honeypot outside the KFUPM network, so that if any malicious activity takes place, the university network will not be affected by it. The motive behind having the Honeypot outside the KFUPM network was to receive Internet traffic directly (unfiltered and unaltered), rather than coming through a firewall and a NAT router. For scenario 2 (see Figure 2), we placed the Honeypot in the faculty housing connected to the Internet through the ADSL network. During our initial runs, the ADSL network was not placed behind the firewall and at that time we recorded activities of certain viruses that had been active within the network.

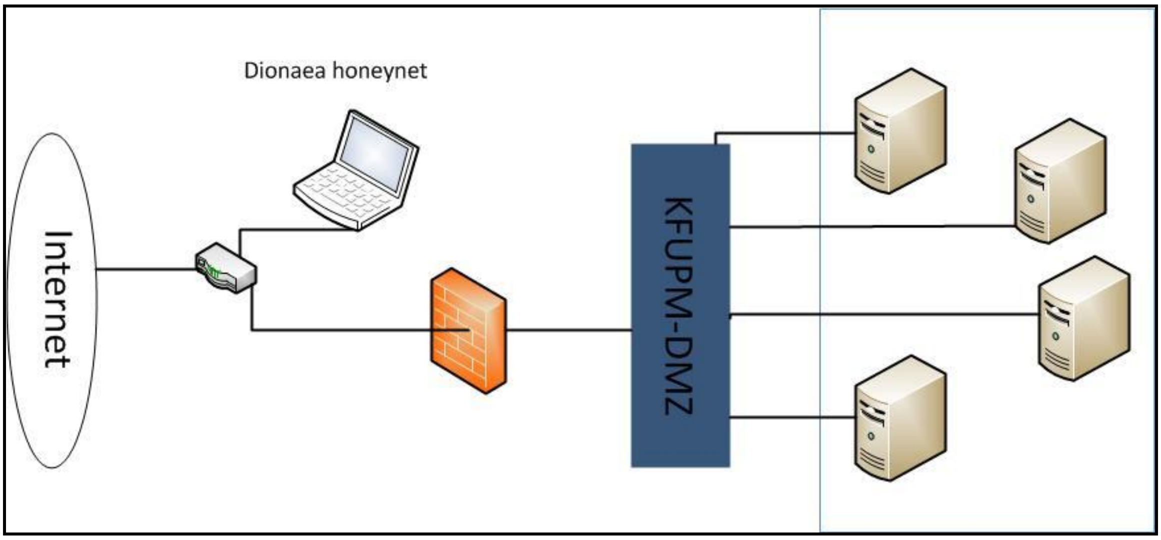


Figure 1. The Pilot deployment map on the KFUPM network

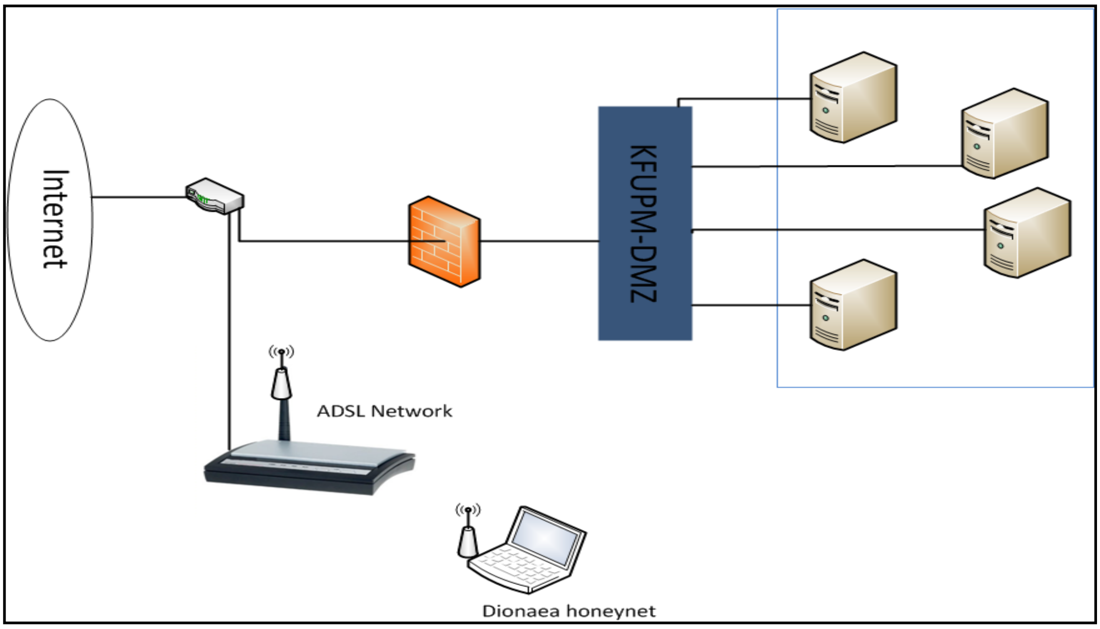


Figure 2.The ADSL deployment scenario for the pilot run

2. Types of Activities Seen on the Honeypot on November 6th and 9th, 2010

In this section, only a summary of activities is presented. The details about such activities can be found in the detailed report of the Honeynet pilot run. And for the sake of brevity, we only present in this summary report statistics from November 6th, 2010. The trend is similar for November 9th, 2010.

2.1. Summary of activities

November 6th, 2010 – Saturday 11.00AM to Sunday 9.00 AM

1. SIP port scanning – IP: 216.55.161.16, 218.61.234.246, 221.231.150.67, 202.5.168.213
2. IP from china trying (attempting with many passwords) to log into the MS- SQL service offered by Dionaea. There were ~300 SQL login attempts made by the attacker source IP 220.168.169.100. More details about these attempts are included in the detailed report.
3. Port scan from ADSL network from IP 196.15.58.160.

November 9th, 2010 - Tuesday 10.00 AM to Wednesday 10.00 AM

1. Sunday (sundayddr) SIP scanning worm
2. Phpmyadmin attack
3. Vulnerability Scanners

2.2. Protocol Distribution – Packets (6/11/2010)

An Overview of the protocol subdivisions at different layers based on total packets has been included in the detailed report. The total capture window: 11/06 10:58:44.511763 - 11/07 8:51:05.511763 (21 hours 52 mins 21 secs at 10 mins). This included the total packets aggregated by transport layer protocol, e.g. TCP, UDP, and ICMP. Then, it included the total packets aggregated by TCP ports, e.g. HTTP, POP3. It also included the total packets aggregated by UDP port, e.g. DNS, DHCP. Finally, the IP host conversations graph has been reported which shows, for instance, the IP from China attempting to login into SQL service. For the sake of brevity of this report, we have omitted these graphs.

2.3. Connections by Country (6/11/2010)

On November 6th, 2010, we have seen connections coming from 26 different Countries, i.e., Australia, Brazil, China, Colombia, Czech Republic, Egypt, Estonia, Europe, France, Korea, Malaysia, Netherlands, New Zealand, Panama, Peru, Russian Federation, Saudi Arabia, Singapore, Spain, Taiwan, Thailand, Ukraine, United Arab Emirates, United Kingdom, United States, and Vietnam. A sample of such connections is provided in Table 1, while the detailed list of connections is included in the corresponding detailed report.

Table 1. List of Connections

Address	Packets	Bytes	Tx Packets	Tx Bytes	Rx Packets	Rx Bytes	Country
220.168.169.100	15220	1191553	5312	476177	9908	715376	China
212.26.1.95	159087	14110522	27571	2066247	131516	12044275	Saudi Arabia
200.35.150.115	125	8726	81	5750	44	2976	Panama
91.197.129.127	481	33834	321	22794	160	11040	Ukraine
212.138.69.17	86	20464	69	19206	17	1258	Saudi Arabia
67.228.44.10	104	6380	73	4498	31	1882	United States
121.78.119.144	3	184	2	122	1	62	Korea
110.232.114.233	12	712	6	388	6	324	Australia
125.230.145.36	39	2497	24	1639	15	858	Taiwan
94.23.236.197	1	453	1	453	0	0	France
211.234.125.69	3	239	2	150	1	89	Korea
194.109.20.90	1	60	1	60	0	0	Netherlands
222.154.105.248	2	128	1	74	1	54	New Zealand
60.50.67.55	2	128	1	74	1	54	Malaysia
178.94.188.173	2	128	1	74	1	54	Ukraine

3. Analysis of Observed Network Traffic Behavior

On our deployed Honeypot, the following ports were kept open for communication: 21, 80, 135, 445, 1130, and 1433.

The TCP ports that were kept open were subject to penetration by malicious codes. As can be seen from Figure 3, port 1433 had the highest number of hits (54000) and port 445 was second with a hit count of nearly 400. It should also be noted that some of these penetration activities were launched by the SAHNET team to test the Honeynet.

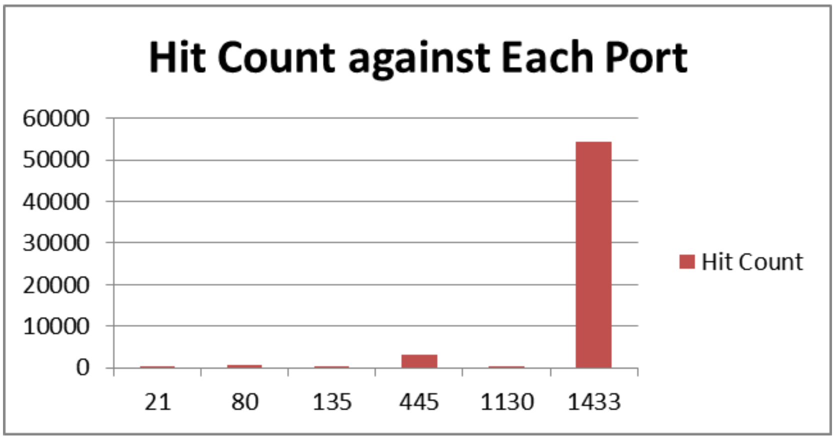


Figure 3. Analysis of the hit counts against the open TCP ports

In Figure 4, we illustrate the most active host IP addresses, which attempted to communicate with the deployed Honeypot.

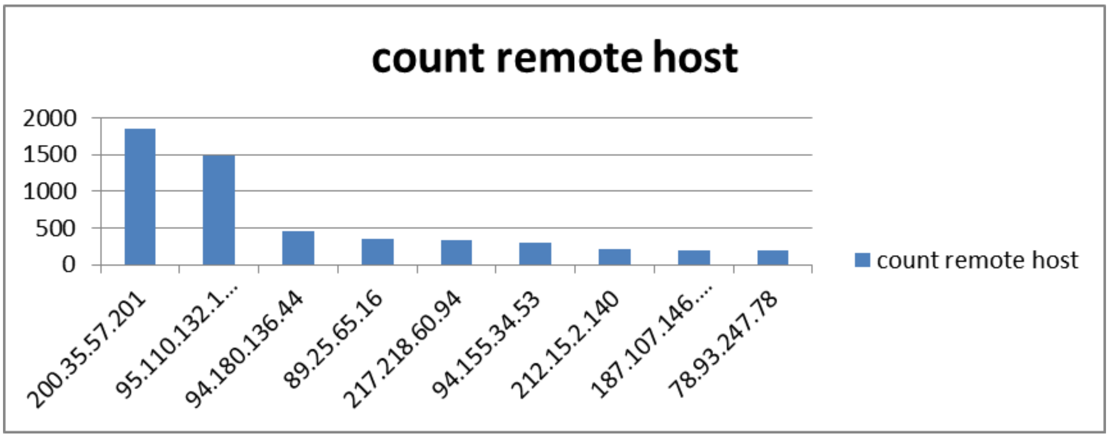
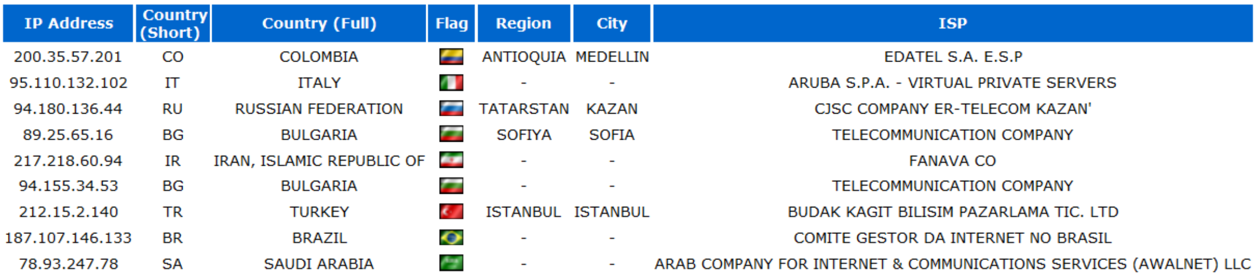


Figure 4. Host count of the source addresses attempting unlawful access to the Honeypot.

From Figure 5, it is evident that several countries from across the globe had host machines attempting to access the Honeypot deployed at ITC.



IP Address	Country (Short)	Country (Full)	Flag	Region	City	ISP
200.35.57.201	CO	COLOMBIA	🇨🇴	ANTIOQUIA	MEDELLIN	EDATEL S.A. E.S.P
95.110.132.102	IT	ITALY	🇮🇹	-	-	ARUBA S.P.A. - VIRTUAL PRIVATE SERVERS
94.180.136.44	RU	RUSSIAN FEDERATION	🇷🇺	TATARSTAN	KAZAN	CJSC COMPANY ER-TELECOM KAZAN'
89.25.65.16	BG	BULGARIA	🇧🇬	SOFIYA	SOFIA	TELECOMMUNICATION COMPANY
217.218.60.94	IR	IRAN, ISLAMIC REPUBLIC OF	🇮🇷	-	-	FANAVA CO
94.155.34.53	BG	BULGARIA	🇧🇬	-	-	TELECOMMUNICATION COMPANY
212.15.2.140	TR	TURKEY	🇹🇷	ISTANBUL	ISTANBUL	BUDAK KAGIT BILISIM PAZARLAMA TIC. LTD
187.107.146.133	BR	BRAZIL	🇧🇷	-	-	COMITE GESTOR DA INTERNET NO BRASIL
78.93.247.78	SA	SAUDI ARABIA	🇸🇦	-	-	ARAB COMPANY FOR INTERNET & COMMUNICATIONS SERVICES (AWALNET) LLC

Figure 5. Country-wise distribution of the source IP addresses of communicating devices with the Honeypot.

In Table 2, we enlist a sample of the web locations and file names from where our Honeypot was instructed to download malware, by the penetrating processes/codes. The detailed list is included in the detailed report. This phenomenon occurred whenever some infected machine/attacker tried to infiltrate the Honeypot to instruct our Honeypot to download the malware.

Table 2. Host machine IP addresses from where the Honeypot was instructed to download malware, post successful penetration

Occurrences	Malware file names and locations
24	http://74.63.78.13/our90.exe
10	http://208.53.183.250/wshh.exe
9	http://208.53.183.250/tensa.exe
8	http://208.53.183.181/M.exe
7	http://208.53.183.171/our90.exe
6	http://208.53.183.181/F.exe
6	http://208.53.183.181/c.exe

Table 3 enlists a sample of the IP addresses of the infected/attacker machines which attempted to infiltrate our Honeypot and instruct it to download a piece of malicious code from the Internet. The detailed list is included in the detailed report. The hash of the files downloaded is also added to the table to show an IP address-to-malware mapping. While some of these malwares were successfully downloaded to our Honeynet, others failed. In the table, we added the actual name of the malware that was successfully downloaded, and the synonyms for that malware. Furthermore, we also list the day of capture of all malwares by our Honeynet, as well as the date of the 1st notice of each malware based on information from www.Prevx.com.

Table 3. List of IP addresses that instructed the Honeypot to download malwares

Count	Hash of Malware	Remote IP	Name	Other names	First Seen Prevx.com	Captured
6	116ea24df855b032 2d7845364f27dd49	196.15.56.92	ouyr.exe, rzri.exe, tegr.exe	syscr.exe, mzrh[1].exe, 67.exe, 30.exe, 64.exe, 03525389.dat	India on Nov 26 2010	27/11/2010
7	02f360845cb765a3 7313f27ad68f41ba	196.15.56.90	bllss.exe	syscr.exe, 47.exe, 76.exe, 37.exe, 71832714.dat, 71.exe, 50.exe, 45.exe, 67422406.exe	Chile on Nov 22 2010 Egypt on Dec 7 2010	23/11/2010
4	cb6ec94b76c5d80f3 dbe5140ea36d312	196.15.56.46	m0bis.exe	syscr.exe, 72.exe, 55.exe, 38397365.exe	Mexico on Nov 24 2010	24/11/2010
3	401ee433272a4cdc 25d058cde312a5ca	196.15.56.26	c.exe	05358013.dat, 20588038.dat, syscr.exe, c[4].exe, c(1).exe, 23971242.dat, 71.exe,	Morocco on Nov 16 2010	17/11/2010

11	339a4f18757fc4a5a 337cb290aff4975	196.15.51.81	f12g33.exe, tensa.exe	http://www.prevx.com/filenames/268708006321035720-X1/TENSA%5B1%5D.EXE.html	Colombia on Dec 1 2010 India on Dec 1 2010	2/12/2010
7	775af7a2d1b54be8a f29d4647b395e1f	196.15.51.81	bdnu.exe, pcxd.exe, mmboo.exe	syscr.exe, bdnu[1].exe, uqlc[1].exe, wnzc[1].exe, pcxd[1].exe, bdnu.exe, 74.exe, 27.exe, 05.exe, 67.exe	Pakistan on Nov 26 2010 Mexico on Nov 26 2010	26/11/2010

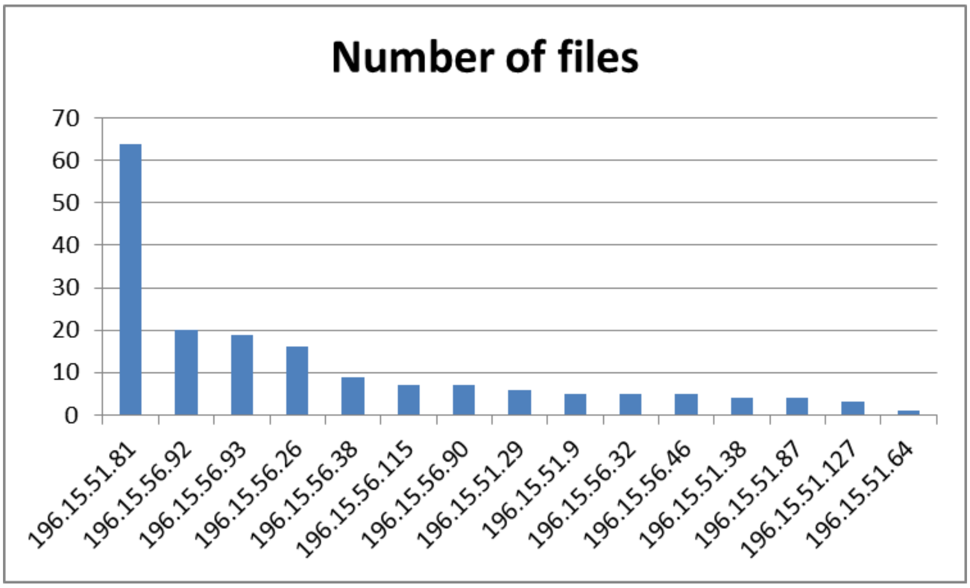


Figure 6. Number of file downloads requested by each hacker IP address listed in Table 2.

4. Dionaea Commands used for Analysis

Dionaea stores all logs in a sqlite database. To extract information from the database, we need to run a diverse range of SQL commands, which are omitted in this summary report for the sake of brevity. The details are available in the detailed report.

5. Analysis of the MS08-67 Worm

The most common attack which we faced during our pilot run was the MS08-067 exploit-based worm. The MS08-67 exploit is a very serious backdoor in the Windows file sharing feature. This can enable any remote machine to execute instructions on the targeted machine. In our case, the instruction to the Honeypot was always to download a malware file from a given web address. The exploit makes use of the Windows file sharing service, which runs on port 445. It starts with two commands to establish the network and transport layer connections necessary to handle the exploit:*connect()* and *smb_login()*. Then, the module begins to build the actual attack string. An important practice that these exploit writers tend to use is the randomization of any non-static portion of the attack. This technique ensures that static content signatures cannot be used to detect the attack. The module then sends the attack over the SMB connection. The attacker program creates a malformed NetPathCanonicalize packet, which actually writes 700 bytes into the target machine which in our case is the instruction requesting to download malware. We extracted these 700 bytes from the trace using Wireshark. Then, we used a tool provided by Dionaea to decode these 700 bytes. In this example, *sample1.bin* is the binary file containing these 700 bytes. This decoding portion contains the URL for the malware download location. Our machine will go to this location and will download the malware. More details on how this exploit works is included in the detailed report.

6. Analysis of Virus Behavior

Two different viruses have been captured by the Honeypot for analysis, a 92KB virus and a 96KB virus. When the 92KB virus is installed in a system, it tries to connect to the DNSserver "ms.mobilerequests.com". For testing purposes, we placed another machine with the same DNS name, then the virus tried to send data on the UDPport1863,of length 7 bytes each time. The data comprised of the following HEX strings: "616a61f3eae3a8" "613e5ba7d0b792" "611d06848d94cf" "6112228ba99beb". No visible pattern was noticed. A similar behavior is seen for the 96KB virus and is included in the detailed report. There were also some effects seen on the registry that were caused by the captured viruses, in order to create their backdoors or change the view of the virus and these have been included in the detailed report.