Assessing the Security Health of Public WiFi Environments Using Mobile Devices

Aysha A. Al Kharoossi

MSc. Thesis

June 2016

A thesis submitted to Khalifa University of Science, Technology and Research in accordance with the requirements of the degree of M.Sc. in Information Security
Assessing the Security Health of Public WiFi Environments Using Mobile Devices

by

Aysha A. Al Kharoossi

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science in Information Security

at

Khalifa University

Thesis Committee

Dr. Khaled Salah (Supervisor & First Examiner), Khalifa University

Dr. Joonsang Baek (Co-Supervisor & Examiner), Khalifa University

Prof. Ernesto Damiani (Committee Chair), Khalifa University

Prof. Ernesto Damiani (Second Examiner), Khalifa University

June 2016

KHALIFA UNIVERSITY
I. Abstract


In recent years, a remarkable increase has been observed in cyber attacks targeting smartphone and WiFi networks. These attacks occur due to the wide use of smartphones and the vast amount of data transferred over Wi-Fi enabled technology. The main problem is that users tend to trust free public wireless networks for sending emails, logging in to a variety of accounts and even for performing online banking, without being aware of whether the wireless network is safe or being used as a platform by attacker to launch attacks to compromise mobile devices of legitimate users. In legacy system networks, the presence of malicious activities is identified by implementing Intrusion Detection Systems (IDSs). However, it is not easy to apply such a technique within mobile environments. Therefore, low interaction honeypots work as a cost-effective and fast alternative to monitor WiFi networks with the use of just smartphones. Such smartphones when equipped with honeypots are considered as an advanced security monitoring and detection tool that provides the users with an early assessment for the network security health.

There are many free public WiFi networks available for users in malls, airports and cafes where users can directly connect and perform different online activities without knowing how secure the wireless environment is. Many questions arise. Are there any attacks performed over these network and what are the types of these attacks? Can the security be assured in the wireless environment? All of these questions should be a concern by every WiFi user before connecting to a public WiFi network. All of the mentioned concerns and questions are studied in this research work. HosTaGe honeypot-To-Go is selected in this research work to be deployed in Android smartphone, and its efficacy in assessing the security of WiFi networks is studied.

Indexing Terms: < Honeypot, WiFi Security, HosTaGe, smartphones, malware, public WiFi networks>
II. Acknowledgments

I would like to express my deepest sense of gratitude to.

- **Dr. Khaled Salah** for his continuous supervision, guidance, encouragement and support throughout the project.
- **Dr. Joonsang Baek** for his supervision, assistance and valuable technical support throughout the project.
- **Dr. Ernesto Damiani** for his guidance and exceptional support.

I would like to take this opportunity to thank my colleagues at the university and the staff in the Information Security Department for their support.
III. Declaration and Copyright

Declaration

I declare that the work in this thesis was carried out in accordance with the regulations of Khalifa University of Science, Technology and Research. The work is entirely my own except where indicated by special reference in the text. Any views expressed in the thesis are those of the author and in no way represent those of Khalifa University of Science, Technology and Research. No part of the thesis has been presented to any other university for any degree.

Author Name: Aysha Abdulla Al Kharoossi

Author Signature: ______________________________

Date: ______________________________

Copyright ©

No part of this thesis may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, without prior written permission of the author. The thesis may be made available for consultation in Khalifa University of Science, Technology, and Research Library and for inter-library lending for use in another library and may be copied in full or in part for any

Aysha Al Kharoossi
bona fide library or research worker, on the understanding that users are made aware of their obligations under copyright, i.e. that no quotation and no information derived from it may be published without the author's prior consent.
IV. Contents

I. Abstract ................................................................................................................... iii
II. Acknowledgments .................................................................................................. iv
III. Declaration and Copyright ............................................................................... v
IV. Contents ............................................................................................................. vii
V. List of Figures ......................................................................................................... ix
VI. List of Tables ....................................................................................................... xi
1. Introduction ......................................................................................................... 12
   1.1 Overview ......................................................................................................... 12
   1.2 Introduction .................................................................................................... 12
   1.3 Mobile Threats ............................................................................................... 14
      1.3.1 Classification of mobile threat ................................................................. 14
      1.3.2 Most Targeted Mobile Platform in 2014 ................................................ 16
   1.4 Honeypot Approach ....................................................................................... 17
   1.5 Problem statement ......................................................................................... 20
   1.6 Research Challenges .................................................................................... 21
   1.7 Objectives and desirable outcomes ............................................................... 22
   1.8 Project Outline .............................................................................................. 22
2. Literature Review .................................................................................................. 23
   2.1 Introduction .................................................................................................... 23
   2.2 New Types of Honeypots .............................................................................. 24
   2.3 Honeypot Data Consumption ...................................................................... 26
   2.4 Configurations of Honeypots ...................................................................... 28
   2.5 Smartphone Honeypots .............................................................................. 30
3. Design and Implementation .................................................................................. 33

3.1 Introduction ......................................................................................................... 33

3.2 Implementation .................................................................................................. 33

3.2.1 Implementation Steps ..................................................................................... 36

3.2.1.1 Remote Procedure Call Attack (RPC) ......................................................... 37

3.2.1.1.1 Remote Procedure Call Attack Result ..................................................... 39

3.2.1.2 Scanning Attack using nmap ........................................................................ 40

3.2.1.2.1 Scanning Attack Result ........................................................................... 41

3.2.1.3 Metasploit Attack ......................................................................................... 41

3.2.1.3.1 Metasploit Attack Result ......................................................................... 44

3.2.1.4 Nessuss Attack ............................................................................................ 45

3.2.1.4.1 Nessus Attack Result ............................................................................... 46

3.3 Performing Attacks in Khalifa campus wireless ................................................. 46

4. Data Collection .................................................................................................... 49

4.1 Introduction ......................................................................................................... 49

4.2 Transferring Data from HosTaGe Database ....................................................... 50

5. Analysis ............................................................................................................... 51

5.1 Introduction ......................................................................................................... 51

5.2 Data Analytic and Intelligence .......................................................................... 51

5.2.1 Add Data to Splunk ......................................................................................... 51

5.2.2 Search in Splunk .............................................................................................. 55

5.2.2.1 Summarized Analysis .................................................................................. 57

6. Conclusion .......................................................................................................... 57

6.1 Summary ............................................................................................................ 57

6.2 Future work ........................................................................................................ 57

7. References ............................................................................................................ 59
V. List of Figures

FIGURE 1 USAGE OF OPERATING SYSTEM IN 2014 ................................................................. 16
FIGURE 2 DISTRIBUTION OF MALWARE IN 2013 BETWEEN DIVERSE OPERATING SYSTEMS .......... 17
FIGURE 3 DOWNLOADING HOS'TAGE HONEYPOT ............................................................... 34
FIGURE 4 ROOTING THE DEVICE .......................................................................................... 34
FIGURE 5 ACTIVATING HOS'TAGE ......................................................................................... 35
FIGURE 6 PROFILING HOS'TAGE .......................................................................................... 36
FIGURE 7 ATTACKING TARGET IN THE SAME ACCESS POINT .............................................. 37
FIGURE 8 IDENTIFYING IP ADDRESS ................................................................................... 38
FIGURE 9 IDENTIFYING IP ADDRESS ................................................................................... 38
FIGURE 10 LAUNCHING RPC ATTACK .................................................................................. 38
FIGURE 11 IDENTIFYING RPC ATTACK ............................................................................... 39
FIGURE 12 DETECTING RPC ATTACK .................................................................................. 39
FIGURE 13 NMAP SCANNING ATTACK ................................................................................... 40
FIGURE 14 HOS'TAGE IS DETECTING SCANNING ATTACK ................................................... 41
FIGURE 15 METASPLOIT ATTACK IN GNACK TRACK .......................................................... 42
FIGURE 16 METASPLOIT ATTACK IN GNACK TRACK .......................................................... 42
FIGURE 17 THE ADDRESS OF METASPLOIT WEB CONSOLE .............................................. 43
FIGURE 18 FILTERING PAYLOADS ......................................................................................... 43
FIGURE 19 LAUNCHING BIND_VNCINJECT_PAYLOAD ......................................................... 44
FIGURE 20 DETECTING METASPLOIT ATTACK ................................................................... 44
FIGURE 21 STARTING BASIC SCAN ON THE NETWORK ....................................................... 45
FIGURE 22 PERFORMING NESSUS ATTACK ....................................................................... 45
FIGURE 23 DETECTING NESSUS ATTACK .......................................................................... 46
FIGURE 24 ZENMAP RESULT IN KHALIFA UNIVERSITY NETWORK ....................................... 47
FIGURE 25 YAS MALL WIRELESS ACTIVATION ................................................................... 48
FIGURE 26 HOS'TAGE LOG FILE ......................................................................................... 49
FIGURE 27 HOS'TAGE EXPORT FORMAT ............................................................................ 49
FIGURE 28 SPLUNK SOFTWARE SIGN IN ............................................................................. 50
FIGURE 29 UPLOAD LOG FILE TO SPLUNK ....................................................................... 50
FIGURE 30 UPLOAD THE FILE FROM LOCAL MACHINE ......................................................... 51
FIGURE 31 REVIEW UPLOADED FILE INFORMATION ............................................................ 52
FIGURE 32 SET SOURCE TYPE ............................................................................................ 53
FIGURE 33 UPLOADED DATA ON SPLUNK SUCCESSFULLY .................................................. 54
Assessing the Security Health of Public WiFi Environments Using Mobile Devices

FIGURE 34 SEARCH BASED ON APACHE SERVER .................................................................55
FIGURE 35 SEARCH BASED FEB MONTH .................................................................55
FIGURE 36 SEARCH BASED ON KU-STAR ...............................................................56
FIGURE 37 SEARCH BASED ON YEAR 2016 ...............................................................56
VI. List of Tables

**TABLE 1** THE NUMBER OF SMART APPLIANCES DISTRIBUTED GLOBALLY IN 2012 AND PREDICTION FOR 2016 BASED ON STATISTICS FROM CANALYS [1] ............................................................................................................. 13
**TABLE 2** CLASSIFICATION OF MOBILE THREATS ............................................................................................................. 15
**TABLE 3** TYPES OF HONEYPOTS ................................................................................................................................. 19
1. Introduction

1.1 Overview

This project explains the functionality of honeypot on Android Platforms of smartphones. It provides an overview of the honeypot and explains the purpose of honeypot deploying and the deployment strategies. Also, two types of honeypot are considered: Low-interaction and High-interaction honeypot. A comparison between them is conducted by providing some examples from each type. Low-interaction honeypot architecture and requirements are focused on. The HosTaGe application is highlighted due to how it works as a convenient honeypot for Android smartphones which detect malicious activities in a wireless network environment. This includes installing HosTaGe application, configuring the application and performing different attacks to make sure that HosTaGe application can detect almost all types of attacks. Also, the numbers of attacks performed on the free public wireless network are studied, as well as the types of these attacks. Furthermore, the smartphone is connected to public wireless networks in crowded Malls for a long period. Also, the collected data in the location is analysed separately based on different aspects. Finally, the thesis is concluded with a summarized analysis of the security of non-authenticated public wireless networks in Abu Dhabi.

1.2 Introduction

Nowadays, smartphones, tablets, and comparable smart devices have become very popular, as they offer a classical cell phone and personal computer pooled in one device. Users who own these smartphones are able to access data easily through diverse network connections such as Wi-Fi and cellular. Accordingly, they can download, install and utilize the most advanced software applications anytime they wish. These smart devices are becoming extremely demanded and this is expected to rise over coming years. Table 1 shows the number of smart appliances distributed globally in 2012, and the prediction for 2016 based on statistics from Canalys [1].
Table 1 the number of smart appliances distributed globally in 2012 and prediction for 2016 based on statistics from Canalys [1]

According to the most recent forecasts from Canalys, smartphones and tablets presented 42% of all appliances shipped in 2012 [2]. Furthermore, smartphones shipment is projected to double to reach 1.3 million appliances in 2016 with annual growth rate of 18%. Although smartphones provide significant advantages to users, they facilitate incessant access for cyber-criminals to sources of personal and financial data of the owners.

Smart devices features accelerate the ability to use mobile applications and network services while at the same time, creating a perfect ground for malicious software. Based on Cisco report, the approximation of monthly disposing mobile data will reach 9.6 exabytes by 2017, which will surpass the world’s population. Because of the vast amount of data that is currently transferred over the wireless network, cyber-criminals have access to diverse sources of data with the arrival of smartphones. The growth of mobile applications that facilitate online payments present another extent to breach meanwhile attackers have created direct monetary advantages from the smartphones' users. In addition, the use of advanced network technologies such as Wi-Fi in smart
Assessing the Security Health of Public WiFi Environments Using Mobile Devices

Aysha Al Kharoossi

mobiles enables the attackers to attack the mobile phone network. Moreover, the rising use of mobile networks presents new exposures to mobile technologies network [3]. Such technologies will certainly be targeted and violated with cyber-attacks that threaten the security and confidentiality of transferred data. Therefore, it is crucial to identify, analyze and address the weaknesses and threats in the advanced network of smartphones and how the users react with these technologies.

In order to reduce possible mobile threats, expected vulnerabilities on wireless networks need to be addressed in order to build a protection prior to the appearance of threats. For this purpose, the honeypot for Android smartphones will identify and observe data of attacks aiming at smartphones through the wireless networks. The purpose of this work is to assess the security of non-authenticated wireless networks in public areas by implementing HosTaGe application, which works as a convenient honeypot for Android smartphones, to identify, examine and recognize the new weaknesses and threats to wireless network of smart mobiles.

1.3 Mobile Threats

Cybercrime is directly linked to cyber threats that are targeting mobile devices through malicious software. Most of this software are created and distributed to target the mobile users specifically through their personnel and financial data.

Smartphones have long been developed as personal and popular electronic devices. These devices can be moved easily from one place to another. At the same time, usability of these devices has grown from isolation to a community of connected mobile devices. Smartphones, tablets, notebooks, media players and many other devices are used as essential components in mechanical and electrical systems. These devices need connectivity to be a part of a large entity of an internet. Due to borderless network connections, these devices are being faced with growing mobile threats. Every mobile device has its own environment, which includes the operating system, hardware configuration requirements, update process and external resources (market place or app store) which vary for different mobile devices.

1.3.1 Classification of mobile threat

This section will briefly focus on the most popular mobile threats that smartphones’ users might suffer from.
Attacks | Description
--- | ---
Man in the Browser (DroidDream, Droid Dream Light and DroidKungFu) [4] | This kind of malicious attack steals financial information through online banking or online transaction. It can be performed by installing a Trojan horse on the user's computer, enabling the criminal to modify real time transactions.
Disposal | Most of the information is stolen through disposed devices, where data is physically present until it is overwritten by other data. Simple format or resetting of devices will not help to completely remove the confidential user information.
Loss and Theft | Compact devices are easy to be lost or stolen. Gaining access to confidential data is easy if no appropriate security measures have been taken.
Spam/mspam (The AdSMS malware) [4] | Unwanted messages are sent using electronic messaging systems. In this act, an unsubscribed or unauthentic material has been received by users to unintentionally charge for money, or breach personal and secure information to the intruders.
Network Connectivity | Majority of attacks fall under this umbrella, where all the attacks penetrate and intercept the user’s confidentiality. Weak access and network security policies or measures allow intruders to breach the security terms.
Multiple identities | Attacker uses different identities to gain access control or to obtain confidential information from their targets.
Software and Application vulnerability | Most of these threats are introduced to the system due to OS and application vulnerabilities, where the developer used dirty and quick solutions to capture market place an

Table 2 classification of mobile threats
1.3.2 Most Targeted Mobile Platform in 2014

Mobile devices have long been used as personal devices for communication and social networking. Kaspersky lab and B2B international in 2014 announced that 77% users use multiple devices to use the internet. According to IDC, the types of platforms of portable devices are as follows:

![Pie chart showing the usage of operating systems in 2014](image)

Figure 1 usage of operating system in 2014

Android platform was the leader among many mobile developers, representing 85% of the market as shown in the diagram. The reasons behind this high demand of the Android operating system are its ease of use and the ability to modify the OS in order to match business requirements without any payment fees. Meanwhile, cybercriminals are also attracted to this operating system to build malicious applications and freely distribute them among such platforms. [5]
Kaspersky Lab reported, in the first half of 2014, 175,442 new Android harmful applications which were distributed. This figure represents an 18.3% increase on the whole year of 2013. [5]

1.4 Honeypot Approach

Honeypot is an innovative mechanism that attracts and lures hackers into a system which appears to be as a real system but is a fake system designed to entice the attacker. The main goal of the honeypot is to detect, analyse and understand the attackers’ behaviour by monitoring their performance and attack techniques on the fake system in order to build better protected systems. Honeypot is a useful method to enhance the security of the system and keep track of future threats [6].

There are many advantages to the deployment of honeypot. Mokube I. and Adams M. [40] stated that honeypots can detect attacks and provide detailed information about each attack if needed. There are attacks that can be solved directly when detected, while others require close examination through monitoring their behaviour in order to provide new and adequate solutions. Honeypot assists when dealing only with malicious traffic, which makes it easier when investigating the attacks.
The main benefit of applying a honeypot is to observe the performance of the attackers, and study the strategies, types and nature of attacks in a specific environment in order to be analysed and finally provide the proper countermeasures.

There are different attacks and malwares, some of which aim to cause financial harm, and others targeting the law enforcement sector. Honeypot helps to identify the security risks and deliver the essential risk mitigation plan, so the more attacks can be captured by honeypot, the more accurate solutions can be provided to counter these risks.

There are two types of honeypot which are based on interaction levels, low interaction level and high interaction level honeypots. High interaction honeypot is an innovative honeypot that has an operating system, so the attacker performs an attack in the manner of compromising a real system. The main advantage of this is that the observer can collect more data from the hacker’s performance. This type of honeypot is high-risk since it offers full access for the attacker without any restrictions. Honeywall and Sebek are decent examples of high interaction honeypots.

The low interaction honeypots imitate the network at the operation level, typically at the TCP/IP layer, and so only require low resources, CPU or memory requirements. Therefore, these honeypots are more convenient for mobile devices. Low interaction honeypots are easy to configure and implement. These types of honeypots are used to capture limited data, such as identifying new worms or viruses and examining the network traffic that the data is going through. In this thesis, firstly the logic of this type will be studied and testing will be conducted for its efficiency. A low interaction honeypot of the Android system smartphone, HosTaGe, will be experimented with. Its version (3.0) has been updated on March 2015. The main purpose of HosTaGe is to provide a quick check of the wireless network for any spreading malware, and assess the security state of the network.
### Types of honeypots

<table>
<thead>
<tr>
<th>Types</th>
<th>Properties</th>
<th>Level</th>
<th>Functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low Interaction Honeypots</td>
<td>Detects the attacks and collects the information about the attackers with known exploits</td>
<td>Intermediate</td>
<td>Limited</td>
</tr>
<tr>
<td>High Interaction honeypots</td>
<td>Fully functional environment to detect new and existing attacks and monitor closely to prevent attacker from using honeypot to target other systems in the environment.</td>
<td>Expert</td>
<td>Fully functional</td>
</tr>
<tr>
<td>Client Honeypots</td>
<td>Any connection to client treated as an attack</td>
<td>Beginner</td>
<td>Very limited</td>
</tr>
<tr>
<td>Server Honeypots</td>
<td>Vulnerable client trying to find malicious servers machine</td>
<td>Beginner</td>
<td>Very limited</td>
</tr>
</tbody>
</table>

Table 3 Types of honeypots

Also, based on the purpose, honeypots can be classified as follows:

- **Production Honeypots**: used to define the motivation and the reasons behind the attacks as well as eliminating the future risk of attacks on the network.

- **Research Honeypots**: used by non-profit institutions that are concentrating on researching the behaviour of different networks.

Honeypot is an effective way to get into the minds of hackers, but it is not always designed to identify hackers. Rather, it allows them to design robust systems as well as guide other experts about these efforts. However, there are fundamental requirements for a mobile low-interaction honeypot. The first of them is visibility, which can be explained as follows: although honeypots normally listen to incoming connections at specific ports that imitate weak services, because of the limitations of resources and security policies of operating systems of mobile appliances, it is difficult to continuously listen to all ports. In order to stop elevating any suspicions in the listened ports, the honeypot should reply based on the specification of protocol.

Another important requirement is the usability of the honeypot. A mobile honeypot application must incorporate a user-friendly interface and simple configuration steps in order to be useful even for ordinary users.
Security and Containment is an essential aspect in honeypot application to strengthen the protection of the smart device. If the security is weak in the user’s device, then the attacker will be able to compromise it and launch any type of attack [7].

For resource utilization, special care must be taken when implementing mobile honeypot to keep away from future overhead. The reason for this is because mobile devices normally have limited power, memory and network bandwidth [7].

Extendibility and Interoperability is the requirement that the honeypot should be able to provide statistics of each attack for the user. Besides that, it must be extendable for generating new protocols for specific vulnerabilities [7].

1.5 Problem statement

Over recent years, smartphone users have tended to trust publically available and free WiFi networks in order to connect and perform their daily online transactions, such as exchanging email and online banking. At the same time, a significant growth of cyber attacks has been noticed, where cyber criminals have taken advantage of users' data availability over mobile devices that are connected to public network to steal information, money and conduct espionage on users. According to Alcatel-Lucent's Motive Security Labs, it was announced in IT Business Edge site that smartphone malicious malware has soared by 25 percent in 2014. Moreover, it was reported that over 15 million mobiles, most of them with Android based systems, are infected. (Sue Marquette Poremba, 2015)

Smartphones have become very popular since they combine a classical cell phone and personal computer in one device. Users who have these smartphones are allowed to install and run the most advanced software applications and use them anytime they wish. Meanwhile, attackers have been attracted to launch attacks on such platforms, resulting in the privacy of users' data stored in smartphones being potentially exposed at any time. People have tended to increasingly use their smartphones for their daily online activities. All data is stored in the smartphones and backed up and updated automatically. When the user authenticates any account using username and password, sometimes it will be saved in a cookie on the browser. Consequently, any hacker can use different techniques to exploit the user's account and steal important data anytime, or can intercept the user traffic and get its credentials. Such threats requires the user to check if the open wireless networks are secure enough to trust. It is important to know how many attacks are performed over Wi-Fi, and the types of these attacks.
Smartphone services have improved lately with the enormous demand in the market. However, the security challenges of these devices and provided services are the main concern of many educated users and security organizations. The main security risks for smartphones are the ease to connect to wireless networks, shared technology issues and availability and leakage of personal data. In classical networks, Intrusion Detection system (IDS) were used to detect and capture malicious activities, whereas with the development of smartphones, it was preferred to implement honeypot as it provides an initial warning on infected network environments.

This thesis will concentrate on assessing the free open wireless network (no security) as many users connect to unknown Wi-fi without considering the risks of the weak security, and without evaluating the security of these networks. Open networks have become more popular nowadays since they offer an inexpensive and fast accessibility to resources. These features also facilitate the ability for the attackers to launch attacks on the network such as Eavesdropping, Session Hijacking and Spoofing and Phishing.

The focus of this project is to assess the security health of public WiFi networks using the honeypot of Android smartphones environment. Our approach for identifying the vulnerabilities and the security challenges for these network environments by deploying and analyzing a honeypot application of Android devices. This tool could play a role in verification of WiFi environment especially in auditing of the security policies in the layer of TCP/IP.

1.6 Research Challenges

There are number of challenges in the work of this project. The first is to install HosTaGe effectively on the Android Smartphone and test its capabilities of capturing different types of attacks. The second challenge is to evaluate HosTaGe efficiency to detect that there are attacks happing between different machines in the same access point in real time. A further challenge is to investigate the security of open public networks in different areas in Abu Dhabi by monitoring the types of attacks that are performed over them. All these data require time and effort to examine and generate reports. The major motivation of this research is to reveal attack information, investigate and provide proper results.
1.7 Objectives and desirable outcomes

The main goal of this project is to examine the security of unauthenticated public wireless networks by fulfilling the following sub-objectives:

- Configure and customize the Android low-interaction honeypot application (HosTaGe) by deploying it on an Android Smartphone and test its functionality by performing different synthesis but real attacks to check its response to this violence.

- Perform many types of attacks and analyze the collected data.

- Investigate the security of public WiFi networks in different areas in Abu Dhabi by monitoring the types of attacks performed over them.

After achieving the defined objectives for this project, an in-depth overview of how secure the open Wi-Fi will be provided, along with the associated advantages and disadvantages.

1.8 Project Outline

The thesis is organized as follows:

Chapter 1 provides introduction and background on security of public network, understanding the problem and the goal of the project.

Chapter 2 presents related work done in the field of honeypot environments in general; in particular, it presents information about high interaction honeypot and low interaction honeypot and examples of some related applications.

Chapter 3 introduces the design, requirements and implementation of the HosTaGe application. The chapter gives steps on how to download and install the HosTaGe on an Android smartphone and configure it in order to start collecting data.

Chapter 4 focuses on attacking the wireless network with different types of attacks and checking the data collection process in HosTaGe on the smartphone. Also, it presents detailed analysis of the output of HosTaGe.

Chapter 5 focuses on testing the performance of HosTaGe application in public areas. This will consider connecting the smartphone for a long period to public wireless networks in an airport or crowded mall, and then analyze the data collected in each
location. Also, these gathered data is observed individually and compared with all other locations.

Chapter 6 will offer conclusion with a brief analysis of the of the public wireless networks security in Abu Dhabi. The challenges, recommendations along with future work will be presented.

2. Literature Review

2.1 Introduction

In 1989, Clifford Stoll published his experience of having found a computer hacker who was searching for secrets in his corporation. He then decided to learn how this criminal gained access to the system in order to trace back the origin of the cracker. The idea was to make a network with forged documents and use a monitoring tool to track the origin of the hacker. However, Stoll did not use the word "honeypot" for his trap system [8]. In 1991, Bill Chewick released his publication "An Evening with Berferd" where he and his colleagues built a trap to follow the location of the cracker and learn his techniques [9].

In 1997, a deception toolkit was introduced with first the honeypot structure, where the idea of this tool is to attack back using deception technique. After a year, in 1998, a new free application, working under Windows operating system, was released. Most people were attracted to use BackOffice-friendly honeypot software because of its ease of configuration and it being free of charge [6]. After this introduction, people became more exposed to this new technology, especially after Lance Spitzner had founded his honey-net project in 1999. In addition to this, “know your enemy” papers were published, giving consumers an in depth understanding of the goal of the honeypot.

In 2002, researchers and companies all over the globe started to use the honeypot in their system to capture malicious software and enhance their security level [6]. Since then, researchers have been employing honeypot extensively for many purposes. In this section, a detailed look at the different categories of honeypot is presented. Many
studies have been published since 1990 until recently, with each research having a certain goal and focusing on a specific target.

This chapter focuses on categorizing these researches based on their goals. Literatures which aim at the evolutions of new types of honeypots are firstly elucidated. Following this, a brief summary of literatures which focus on how to develop consumption of data from honeypots is presented, followed by studies concentrating on the configurations of honeypots. Finally, papers proposing honeypots compatible with smartphones are examined, which is a new era in the security environment.

### 2.2 New Types of Honeypots

There have been remarkable changes in the internet services, such as the enormous adoption of wireless devices, high-speed technology, growth of network application and the variety of users' demography that use the internet for diverse purposes. To cooperate with all these changes, many modern kinds of honeypots have been proposed and published.

First of all, Adachi and Yoshihiro have proposed a low interaction and high-interaction hybrid honeypot called "BitSaucer". The low-interaction honeypot aims at attaining minimal resource requirements, where the goal is to simulate complete replies and responses from the high-interaction honeypot. They run a proxy in each physical host. This proxy generates virtual hosts dynamically besides redirecting network traffic [10]. Every virtual host imitates a complete system in high-interaction honeypots. This solution reduces the resource consumption because the honeypot is invoked automatically in the virtual host only when the network traffic that requires such honeypot arrives at host [10].

In 2007, Alberadi et al. [11] focused on providing a solution for Botnets, which present a major risk for the quality of internet services. They tried to detect and observe malicious behaviors of botnets and disable them. Alberadi et al. designed "SHARK", a spy honeypot with an advanced redirection kit. The objective of the study was to redirect outgoing attacks to other honeypots in order to block any malicious activities which affect other production servers. Meanwhile, the attacker would think that bots are corresponding with hosts outer the network, where in fact they were in the same network [11].

In January 2009, Das introduced the idea of "Active Server" to alleviate the denial of service attack. By implementing the AS design; all clients must be authenticated
before gaining access to the production server. The idea was to hide the production server behind a gateway called AS, so every active server works as an entrance to the production server. The purpose of the AS is to authenticate the client before getting an open path to the server. If the client is not authenticated, then the AS behaves as a honeypot to trap the client there [12]. This way the DoS attackers disallow unauthorized access to the server, while the legitimate clients reach the server safely.

In April 2009, Nazario [13] presented a virtual honeyclient called "PhoneyC", which is an emulated client that provides visibility function into novel and complex client-side attacks. The main purpose of PhoneyC is to eradicate the confusion from various malicious web pages, help study malicious HTTP pages, and monitor new attackers' techniques. They accomplished their goal by combining an active client-side honeypot with an automatic web client parser specifically for transposing dynamic script content such as JavaScript and VB Script. The combined active client honeypot became "A web Clawser" that visits diverse web servers to reveal any malicious content. The result was that PhoneyC actively detected many malicious contents in the Script during the experiment.

In October 2009, Ghourabi et al. introduced a new approach to protecting routing protocols from being compromised. They built a high interaction client side honeypot called "Honeypot Router" that functions as a router in the network. The deployed Honeypot Router assists in early detection of attacks and monitors the behaviours of the hackers in the system. The proposed honeypot uses existing software tools such as Quagga, which is open source routing software that sends messages to remote routers to check if the routers have been penetrated. It also supports the observer to analyse the sent data to the router and detect the attacks. The researchers mainly tested the honeypot for many purposes such as exploiting RIP and OSPF routing protocols in order to elucidate that the honeypot has the ability to capture attacks targeting the router [14].

Li and Schmiz [15] tried to overcome some dilemmas of anti-phishing solutions which are supported by honeypots. Their implementation was based on integrating the genuine electronic banking system into a honeypot that is prepared with honey tokens and carried by some other types of honeypots. Also, they implemented a phishing detector, which would automatically detect any attempt from the phisher to steal money from an account and directly ask the account owner's reconfirmation. Their proposed honeypots called "phoneybots" were aimed at observing bank operations and
transactions using forged accounts to trap phishingers' malicious activities. The process was to accept the request of phishingers dynamically, reply to phishing sites and analyse their possible risk [15].

Prathapani, Santhanam, and Agrawal [16] focused on detecting black hole routers in Wireless Mesh Networks "WMN" by proposing a smart honeypot based on detection system. Their design mainly detects these holes in WMN that have more than one path among two different routers in the network. The researchers provided a solution that depends on three modules. The mechanism is to send a message from the router module to the feedback module using a queried routed. The feedback module should send a reply to the router module; if router module does not receive any respond from the feedback module, then the router module considered the tested router as a black hole attacker. If the feedback is received, the alert module will then inform the entire network to evade the attacker.

Alosefer and Rana in 2010 proposed a low interaction client-side honeypot called "Honeyware" for the purpose of revealing any malicious web servers. Alosefer collected 94 malicious URLs in advance, including 10 benign ones, and tested them using Honeyware and Capture-HPC, a high interaction client-side honeypot. He then compared the two honeypots in order to validate Honeyware. He found that the Capture-HPC took 17 seconds to scan each URL, detecting 62 malicious URLs, listing 23 as malicious and remaining 9 are inconclusive. Since Honeyware is a low-interaction honeypot, it requires an external process engine to process the data collected, thus, it took 1 minute for each URL and detected 83 URLs. The researcher concluded in his experiments that in order to fully take the benefits of high interaction and low interaction honeypots, both types should be integrated for future design [17].

### 2.3 Honeypot Data Consumption

When a honeypot is implemented in the system, the collected raw data must be properly analysed. Experts should have skills in network, applications, hardware, and operating systems and might require management skills, as well as some experience in network security. In recent years, many researches have been conducted to automate the data processing and data mining. This section concentrates on some current honeypot related to utilizing honeypot output.

Zhao and Zhang used honeypots to ban the distribution of worms. Their solution was based on a model that mainly predicts worm propagation. This model groups the hosts
depending on their state as vulnerable, infectious or protected to approximate the infection rate of each host. Following this, the appropriate action would be taken to stop the spread of the infection. They divided the network into sections, first with honeypots implemented in the network system in different computers called "hub nodes". In the model, numbers of honeypots were activated using a systematic method to increase the ability to prevent the distribution of the infections [18].

In 2008, Thonnard and Dadier proposed a framework that reveals the pattern of unknown attacks from honeypot raw data. The main goal was to help prevent the security administrators spending an enormous amount of time and effort to examine the raw output data and the quality of attacks pattern received from the raw data. They used a clustering and similarity distance method to solve these problems [19]. They also deployed a graph-based approach to present the relation between the cluster and the calculated similarity distance.

Fairbanks et al. developed a new technique to solve the limitation of Linux Virtual File System (VFS). The problem with VFS is that it is unable to recognize the file name from i-nodes. When i-node elevates a warning that indicates a potential security attack, the analyst can't find a simple way to find the file name of that attack. For this purpose, they used a method to transform the VFS code in a way that enables the Linux kernel to store data called "dentry", where it includes the path from the i-node to the file name inversely [20].

Chen and his assistants used honeypots to merge an intrusion tolerance into network security forensics called "Dynamic Forensics". They attempted to prove that the raw data of the honeypot is still valuable even though the attacks attempted to change the information. The key idea of the proposed solution is the intrusion detection technique that observes any threat in the system. The forensic structure is activated when the risk occurrence is high. Thus, the traffic will dynamically be redirected to the honeypot [21]. In another situation where the attack is propagated above a certain level, the traffic will be stopped and cut-off to prevent any changes happening to the gathered data. The next step would be to extract the signature and examine the attack.

Narvaez et al. focused on the weaknesses of the high interaction honeypots. The study assessed the hypothesis when a honeypot is deployed on a virtual machine. They claimed that when the malware succeeds to recognize VM implementation of honeypots, the high interaction honeypots are unable to recognize websites that are infected with drive-by-download attack. To check the effectiveness of revealing
malicious websites, they built up two systems: one virtual machine (VM) honeypot and a honeypot on a full system. By comparing the two systems, they found that the VM implemented honeypot is positively detecting malicious websites [22].

2.4 Configurations of Honeypots

Tuning honeypots is complicated and can sometimes be problematic. When configuring a honeypot, certain criteria must be kept in mind in order for the trap to work efficiently; such as when, what and whose actions should be observed. If it is not correctly configured then two possibilities might occur: the desired prey will not be attracted, and the honeypot will be left vulnerable to other hackers. The information that is collected by the honeypots must therefore be very efficient and specific, so as to not receive unnecessary data and information. This section will discuss the various ways through which this configuration of data can be completed most efficiently.

Briffaut presented the structural design of spread high-interaction honeypots that observe attackers, in addition to potential abuses and hijacks on high-interaction honeypots, which will dynamically reinstall if illegal changes take place in the honeypots [23]. He proposed this architecture facilitate the trade-off difficulty between low-interaction and high-interaction honeypots. The status of every honeypot implemented in the network environment is monitored periodically using intrusion detection, alongside various tools that reinstall an entire system dynamically in case of any attack detection. The actual challenge in the solution proposed by Briffaut et al. is the techniques needed to update uncontaminated system copy when every system is capable of actually be modified legally. This needs to log every activity in each honeypot in addition to regular re-downloading of high-interaction honeypots, which considerably raises their downtime, which could possibly be another goal of DoS attacks.

Wang presented a technique to automatically allocate various forms of honeypots to various areas in a network [24]. The developed method dynamically regulates high, medium, and low interaction honeypots in the network. The low interaction honeypots catches known attacks meanwhile it works in line with the medium and high interaction honeypots like a complete system to catch and investigate anonymous attacks. The main idea of the solution is to correctly use the formula while distributing the three types of honeypots. To ensure this, the network was divided to different regions: outside the firewall, on the intranet, in the DMZ, and in a sub-network. When
the threats are identified, using the formula on the fly the, the system will automatically set the areas, the level, and the number of honeypots to allocate. The study proved that the system was competent in decelerating down and blocking attacks.

Carroll [25] used some concepts in order to obtain the best positioning of honeypots among the network, and used diverse varieties of systems: normal production systems, fake normal production systems, honeypots and fake honeypots. The authors then presented models which used the best from each of the systems in order to achieve optimum success with regards to attackers and defenders. However, these models showed an unrealistic view of the attacker-defender relationship, hindering their prosperity. Despite this, if their technique is joined with the dynamic honeypots the one designed by Wang [24], the concept could deem useful in the distribution of honeypots.

Chen worked with SQL injection attacks, while offering advice and guidance on setting up honeypots with the purpose of studying and detecting them. One of Chen’s arguments was that the honeypot should be extremely active for all related manners. It was also suggested that many non-productive structures use the server to ensure the appearance of the honeypots were as real as possible. It was recommended that the procedures are monitored and restricted as they could be used to change the system, leading to the idea that the server would be utilized to stop SQL commands from abusing the database, if a proxy among the web and database servers was established. This honeynet could direct all SQL injections to the database, which should be settled with the real data of a high interaction honeypot. It was suggested that honey tokens can be used in order to be traced. Finally, they recommended strengthening the database so that it would be complex to access [26].

Hecker proposed "Honeyd Configuration Manager", which used an automatic implementation of low interaction honeypots based on the requirements exposed by network scans. Using Nmap, detailed information about the system on the network can be collected, including operating systems and ports that are opened. Using the information compiled, the configuration manager is able to deploy and start low-interaction honeypots. Configuration files are used aiming at allowing administrators to locate ports and addresses of honeypots to each honeypot. Also, specifying the services of servers that are imitated at each honeypot [27]. This research resulted in a successful deployment and testing of "Honeyd Configuration Manager". It enables
system administrators and researchers to build honeynets based on real networks rapidly and effortlessly [28].

Kohlrausch examined the gathered data by a honeypot NoAH pilot testbed using Argos. The honeypot used "Dynamic Taint Analysis" (DTA) through marking every byte of received information and tracing its activities in the network using a debugger. DTA was deployed through the use of an integration of Argos honeypots, Snort intrusion detection system, and a debugger. Argos was used to search for unidentified threats, while Snort revealed popular attacks, each of which were marked as suspicious for future tracing by the debugger. It was announced that the greatest complexity faced was that, because of the large amount of elaborated data collected, it was hard to determine the importance and role of each activity. The analysis of data was hard since the output includes data of various unrelated attacks. Finally, Kohlrausch recommended that basing the data output on network flows would be sufficient to alleviate these difficulties [29].

Spitzner developed methods for catching internal threats using honeypots and honey tokens [30]. He suggested that insider threats should be dealt with differently to outsider attacks as they have access to the system. In order to identify these insider threats, the honeypots should be implemented inside the network and should take up any unexploited IP addresses and they must be high-interaction honeypots. In addition, it is not enough to simply hope that the attackers would come across the honeypots. Instead, they must be directed to them to ensure success. Using their purpose against them, as the attackers are looking for usable information, the honeypots will grant them with information, but it will be information that they do not need to know. For example, false business plans or design specifications, serving as honey tokens [30].

### 2.5 Smartphone Honeypots

Recently, there have been significant changes over the internet; the arrival of new applications, extensive implementation of wireless devices and premise of speed technologies to users in different places. Noticeably, smartphone devices became increasingly more powerful and usable. As a result, novel kinds of honeypots have been developed. Mobile honeypot is implemented effectively to prevent any attacks that target smartphone users. This following section will discusses the new current researches that focus on smartphone honeypots.
Freeman et al., in 2009, [31], discussed a way for building the first generation Smartphone honeypot, SmartPot, along with detecting programmed worms by using Honeyd as a low-interaction virtual honeypot. This was done by imitating Windows Mobile 5 and Windows Mobile 6, with the obtainable TCP/UDP ports of each operating system. They faced a major obstruction during the execution of the Honeyd Smartphone honeypot. Despite this, they found that it is possible to design a honeypot that is specifically created to discover existing automated worms. The methodology is to find the available ports on Windows 5 and 6 mobiles using Nmap scanner and then mimicking those ports in Honeyd honeypot [32].

Two years later, in 2011, Mulliner et al. used a real smartphone device instead of an Android emulator to implement HoneyDroid Smartphone honeypot. The main objective of HoneyDroid is to capture online mobile network and malicious applications attacks [33]. The main difference in this experiment is that the Android's hardware interaction is completely controlled because all HoneyDroid related devices are virtualized, therefore Andrioid is not privileged with access to the hardware directly. On the other hand, the main disadvantage is that the malware might stop the attack and leave the honeypot because the HoneyDroid acts differently from the original Android system.

In August 2012, a low interaction server honeypot was designed based on the honeytrap tools and Dioneae. They mainly concentrated on obtaining statistical analysis of attacks. The study found that a few internet domains have placed a particular focus on attacking mobile networks. Smartphones experience similar amount of attacks as a home network device [34].

On November of the same year, a honeypot observing system for mobile communication was proposed. The concept is to combine an active honeypot with a protection system. They examined the communication behaviours of simulated wireless network environments while capturing the smartphone. Moreover, they tested an Android smartphone infected with a malware to study the feasibility and efficiency of the system. The study concluded that using anti-virus applications on mobile devices leads to weak versatility and resource utilization, and low virus detection rate. These drawbacks can be overcome by implementing a honeypot monitoring tool in the system [35].

During the year 2013, studies related to smartphone honeypot began to rise significantly. In January, Wahlisch et al. compared between mobile and non-mobile
attacks by designing a honeypot system that works on standard PC to capture any malicious behaviour in the traffic [36]. They finalized their study by concluding that smartphones are more vulnerable with malicious applications nowadays.

Another study was conducted in May, which discussed the recent attacks against smartphones. Liebergetd et al. introduced the nomadic honeypot that allows mobile network operators to gather threats on smartphones. They applied their honeypot on a Galaxy S2 smartphone, which has an Android mobile operating system. They concluded that there were some usability drawbacks, such as the battery not lasting long given the computational overhead of the honeypot [37].

On July of the same year, Gelenbe et al proposed a design of NEMESYS honeypot to gather and examine the nature of data of cyber-attacks that targeted the smartphones and the core network in order to combat abnormal behaviours. They developed a data collecting, virtualization and analysis infrastructure, as well as introducing visual analytic technologies for the removal and appearance of large amounts of mixed information of smart mobile systems [38].

This phase we reviews surveys on different types of honeypot implementation and researches. In this field, we have focused on four areas which are new types of honeypots, Honeypot Data Consumption, configuration of honeypots and smartphones honeypots.
3. Design and Implementation

3.1 Introduction

HosTaGe is a Linux-based application that detects malicious activities in a wireless network environment. It checks the wireless networks for spread malware and attacks to provide an immediate evaluation on the security state of that network to the user. The most important step in order to test the security effectiveness of any application is to deploy it and try to examine its response to different types of attacks. This section introduces the setup used in the experimental of HosTaGe; the main goal of this phase is to ensure that HosTaGe application can distinguish most types of attacks.

3.2 Implementation

HosTaGe Honeypot works as a convenient honeypot for Android based smartphones. It is easy to set up and correctly configure. Additionally, its detection capacity for attackers in a shared network is quite effective. Upon activation, it immediately checks for any presence of malware in the shared network and it quickly sends out to all the devices connected to the network an evaluation of the security status in the network. To receive this information, one must have HosTaGe installed.

In this test, HosTaGe is installed in a smartphone as shown in Figure 3. In order to fully use the application, access to the Android file system should be obtained by rooting the device using root checker application and installing Portbinder, as shown in Figure 4. After that, the application is run and activated as in Figure 5.
Assessing the Security Health of Public WiFi Environments Using Mobile Devices

Figure 3 Downloading HosTaGe Honeypot

Figure 4 rooting the device
The application profile is then configured as XP, as shown in Figure 6. The attacker will recognize that the device has vulnerabilities in XP and can be exploited. Furthermore, Profile option provides the ability to choose multiple types of profiles or create a new one. This option is useful to trap attackers by allowing them to assume that the device has weaknesses in a specific service, whereas in reality there are fake weaknesses used to monitor the performance and the activities of the attackers while accessing the targeted system.
3.2.1 Implementation Steps

The steps and requirements followed to install and setup an environment that help to detect attacks are as follows:

There are many clients that have been connected to the same wireless Access Point (AP), as shown in Figure 7, such as XP machine and Linux machine (VM). These machines will launch attacks to the target device (smartphone that downloaded HosTaGe). A Linux Machine is running a Gnack Track and a windows XP SP3
machine has RPC exploitation on it and a smartphone with the following specifications:

- Device: Galaxy S DUOS 2
- Model-Number: GT-S7582
- RAM: 0.75 GB
- ROM: 4 GB
- OS: Android 4.2.2 (Jelly Bean)
- Linux-Kernel: 3.4.5-2712851

The next step is to perform some attacks and check if HosTaGe is able to detect them and ensure the health of the wireless network. In this test, RPC, nmap, Metasploit and Nessus attacks are performed on the network, and the results are checked.

### 3.2.1.1 Remote Procedure Call Attack (RPC)

Remote procedure call attack is a very popular vulnerability in windows XP which leads to a full compromising of the targeted computer. For this examination, first, the target’s IP address must be identified. This can be done by pressing on the phone’s setting, about the device, status and then the IP address will be shown, as displayed in Figure 9. After that, Windows XP machine launched the RPC attack (which is a very
popular vulnerability in Windows XP). The IP of the target has been entered and exploitation has been initiated, as shown in Figure 10.

Figure 8 Identifying IP address

Figure 10 launching RPC attack
3.2.1.1 Remote Procedure Call Attack Result

After launching the attack, HosTaGe identified and detected the attack successfully as shown in Figures 11 and 12.

![Figure 11 identifying RPC attack](image1)

![Figure 12 detecting RPC attack](image2)
3.2.1.2 Scanning Attack using nmap

The second attack is an intense scanning attack using nmap. GnackTrack Linux machine is used to run Armitage tool, as shown in Figure 13. This tool executes an intensive port scan in the network to examine the ability and effectiveness of the application to a massive port scan. This can be done by clicking on the Hosts tab, add hosts, identify the target IP, click on the hosted machine, then Hosts, Nmap Scans and intense Scan.

![Figure 13 Nmap scanning attack](image)
3.2.1.2.1 Scanning Attack Result

HosTaGe was able to identify and detect the scanning attack on the network as shown below in Figure 14.

![HosTaGe is detecting scanning attack](image1.jpg)

3.2.1.3 Metasploit Attack

The third attack is Metasploit attack. Metasploit is a tool used in Linux Gnack Track that performs various exploit codes to a specific weakness against a remote target. This activity results in gaining full access to the victim machine. This attack can be done as shown in Figure 15, by clicking on the application tab, Gnack Track, penetration and framework2-Msfweb, Framework version2.

The following step is to click on application, internet, and Fire Fox web browser as in Figure 16. Then, the address is inserted as shown in Figure 17. After that, it is important to filter the payloads and launch the required one based on the weakness, as shown in Figure 18.
Assessing the Security Health of Public WiFi Environments Using Mobile Devices

Figure 15 Metasploit attack in Gnack Track

Figure 16 Metasploit attack in Gnack Track
Assessing the Security Health of Public WiFi Environments Using Mobile Devices

Figure 17: The address of Metasploit web console

Figure 18: Filtering payloads
3.2.1.3.1 Metasploit Attack Result

After launching Bind_vncinject payload to the target as in Figure 19, HosTaGe was able to identify and detect the scanning attack on the network as shown in Figure 20.

Figure 19 launching Bind_vncinject_payload

Figure 20 detecting Metasploit attack
3.2.1.4 Nessus Attack

The fourth attack is Nessus attack that provides the ability to perform a basic scan on the network. Nessus attack can be done by first logging in to Nessus website, downloading Nessus on your machine, registering for activation code, creating policy and then beginning the scan as shown in Figure 21 and 22.

Figure 21 starting basic scan on the network

Figure 22 performing Nessus attack
3.2.1.4.1 Nessus Attack Result

HosTaGe was able to identify and detect the Nessus attack in the network as shown below in Figure 23.

![Figure 23 detecting Nessus attack](image)

3.3 Performing Attacks in Khalifa campus wireless

Khalifa University is a learning and research institution. It has more than 1300 students in different engineer programs. The test is performed on HosTaGe in Khalifa University campus for one week. The device that installed HosTaGe is a SAMSUNG smartphone, which was connected to the WiFi of the campus and nmap scan was performed for the full week. The results showed that the honeypot application detected all attacks successfully. The next step was to attack the Wi-Fi in campus by many volunteers based on specific instructions.

A USB Flash was prepared that contains Zenmap application, which is GUI version of nmap scan. Zenmap was used to scan the wireless network as shown in Figure 24. The following steps were given to a number of student volunteers to initiate malicious traffic.

- Make sure that you have download WinPcap on your laptop from [www.winpcap.org](http://www.winpcap.org)
- Connect your laptop to “KU-AUH-STUDENT” using your ID and password.
Make sure that SAMSUNG Phone is connected to “KU-AUH-STUDENT”.
Insert the USB flash.
Copy the "Nmap" folder to your desktop.
Double click on the "nmap" folder on your desktop.
Double Click on Zenmap.exe file.
Type the IP address of the target “10.10.131.98”.
You can check the IP address of the target (SAMSUNG phone) by pressing on setting→ about device → status→ IP address.
Choose the “intense scan” in the profile option.
Click on Scan.

After following the above steps, the network would be scanned and HosTaGe should detect the scanning attack. They scanned the network from different laptops which each laptop scanning at different time. All scans were detected successfully by HosTaGe.
The following step was to test the public wireless network environment of Yas Mall. Initially, the university sent a request to Yas Mall's management in order to provide a continuous public wireless connection for full week, to which they agreed and took the MAC address of SAMSUNG phone (targeted device) to give the privilege of accessing Yas Mall Wi-Fi for one week.

The phone was connected and checked on a daily basis. The results showed that there were no attacks during week days and weekends. Therefore, Yas Mall is considered to be a safe environment as shown in Figure 25.

![Figure 25 Yas Mall wireless Activation](image-url)
4. Data Collection

4.1 Introduction

An important component in HosTaGe is hostage _default.log, as it records and logs all captured attacks with the timings along with the attacks’ types in the database. Once SAMSUNG phone is connected to the Wi-Fi and HosTaGe is activated and running, it will start detecting attacks and then store all data into the specified database. The following section will illustrate log file in detail and the captured data.

Figure 26 HosTaGe log file
4.2 Transferring Data from HosTaGe Database

The database file is stored and exported from the device to be able to analyze the real data detected in the log file. The exported file is used to observe the content of the database. Splunk software was used to demonstrate the content of the database with the ability to search through results. The data in the HosTaGe application can also be saved and exported into diverse formats such as plaintext and JSON. The application provides the options in order to divide the fields of the attack dates, as shown in Figure 27 below. In this project, data were exported in plaintext format and opened in the Notepad application.

![Figure 27 HosTaGe export format](image-url)
5. Analysis

5.1 Introduction

In this section, the collected data detailed in chapter 4 is analyzed in detail. As mentioned earlier, there are different attacks performed in various free wireless networks in order to check the validity of the Honeypot application using numerous techniques. The methodology of the analysis is to examine the collected data from each type of attack in different wireless environment. The tool used for analysis methodology is also explained. Guided steps of data processing are demonstrated.

5.2 Data Analytic and Intelligence

There is a need for an automated tool in this stage to analyze collected data efficiently and quickly. There are many tools available in the market to analyze data; however, Splunk software, as shown in Figure 28, was chosen as the best software in 2013 for security information and event management. For this reason, it has been chosen to study and examine all gathered data. Splunk is software that provides the ability to search, examine and study information from many applications, server or network logs.

5.2.1 Add Data to Splunk

Different varieties of data can be exported to Splunk software such as syslog, windows logs registry and windows performance metrics. For the collected data in this project, the log format file is used. Choose the option to upload the file.

---

Figure 28 Splunk software sign in
Upload the log file to Splunk from the local machine as shown in 29 and 30.

Figure 29 upload log file to Splunk

Figure 30 upload the file from local machine
A new source type can be chosen as in Figure 31, or a previous source type applied based on the format of the file that will be analyzed and then a review of the uploaded file will be displayed as in Figure 32.

Figure 31 review uploaded file information

Figure 32 set source type
After completing all stages from uploading the file, select source type, set source type, input settings and review, the user will be confirmed that the file has been successfully uploaded as in Figure 33. Splunk provides several selections on how to analyze and monitor data. As an example, the user can attach data from a file or directory, so that every time Splunk is accessed it will dynamically update the uploaded file. Moreover, it provides the option to create a whitelist or blacklist, translating into data that should or shouldn’t be monitored by Splunk for this specific source.

![Figure 33 uploaded data on Splunk successfully](image-url)
5.2.2 Search in Splunk

The search option in Splunk’s Dashboard can be used to analyze specific data as shown below in Figure 34.

![Figure 34 search based on Apache server](image)

The output of the search is based on defined content of the data and fields, as shown in Figure from 35 to 37 below. So, it examines the data in every field based on the

![Figure 35 search based FEB month](image)
shared factor.
5.2.2.1 Summarized Analysis

In this project, we first concentrated on checking the functionality and proper operation of the HosTaGe by performing some attacks to provide an accurate result. From all collected data, HosTaGe was proven to be effective and successful application in detecting many attacks that take place in the wireless network. This information about the attacks can be distributed to the community of other HosTaGe’s users by showing the location of the infected wireless, the type of attacks, percentage of attacks and the services that have been attacked. Therefore, users will be aware and able to logically decide whether or not to connect to the free public wireless network.

6. Conclusion

In this concluding chapter, a brief summary of the project is given with the difficulties that were raised, along with the solution on how to encounter each challenge during the entire project. Also, a list for suggestions and future work is provided.

6.1 Summary

Many smartphone users have the tendency to connect to any available free unauthorized wireless network to perform their daily online transactions. These actions have created many challenges in the security sector and give rise to many mobile threats. This research work highlighted the mobile threats over open WiFi networks by first implementing HosTaGe honeypot-To-Go in an Android smartphone to assess the security health of any WiFi environment, and subsequently performing field tests on real WiFi networks. We studied the efficacy of our approach by collecting real data after deploying our mobile devices in a number of public networks. We then examined the collected data to identify the type of attacks and number of attacks.

6.2 Future work

We carried out real security assessment in only one publically available WiFi netwok which was Yas Mall, Abu Dhabi. Also, the assessment was conducted for one week. The assessment scope might be extended to cover more malls, airports and many cafes.
which will improve and add more value besides affording detailed study about the security of the free public wireless network. Moving forward, and as a future work, we suggest the following:

1. Deploy HosTaGe honeypot in different Emirates in crowded malls for longer period of time and compare the results.

2. Choose educational campuses such as schools, colleges and universities which host more people using the wireless for their activities that hackers find it an attractive place to target and compromise.
7. References


Assessing the Security Health of Public WiFi Environments Using Mobile Devices


Assessing the Security Health of Public WiFi Environments Using Mobile Devices


[40] Mokube, Iyatiti; Adams, Michele, "Honeypots: concepts, approaches, and challenges," in
Assessing the Security Health of Public WiFi Environments Using Mobile Devices


