Mid Semester Project Report

on

Honeypots: Tracking Hackers

Submitted in partial fulfillment for award of

BACHELOR OF TECHNOLOGY

Degree

In

COMPUTER SCIENCE & ENGINEERING

2022-2023

Under the guidance of:

Ms. Anjali Yadav

Submitted By:

Sanju Tomer(2000330109010)

Yugal Teotia(2000330109012)

Mohd Nadir(2000330109006)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

RAJ KUMAR GOEL INSTITUTE OF TECHNOLOGY

5 K.M. STONE DELHI-MEERUT ROAD, GHAZIABAD

Affiliated to Dr. A.P.J. Abdul Kalam Technical University, Lucknow

October 2022

Raj Kumar Goel Institute of Technology Ghaziabad

ISO 9001:2015 Certified

5th KM. STONE, DELHI-MEERUT ROAD, GHAZIABAD (U.P)-201003

Department of Computer Science & Engineering

Project Progress Report

1. Course : Bachelor of Technology
2. Semester : VIIth
3. Branch : Computer Science & Engineering
4. Project Title : HoneyPots: Tracking Hackers
5. Details of Students:

S. No.	Roll No.	Name	Role as	Signature
1	2000330109010	Sanju Tomer	Team Leader	
2	2000330109012	Yugal Teotia	Coder, Report	
3	2000330109006	Mohd. Nadir	Designer, Tester	

6. SUPERVISOR: Ms. Anjali Yadav

Remarks from Project Supervisor:

…………………………………………………………………………………

…………………………………………………………………………………

…………………………………………………………………………………

…………………………………………………………………………………

SYNOPSIS

A honeypot is a network-attached system set up as a decoy to lure cyber attackers and detect, deflect and study hacking attempts to gain unauthorized access to information systems. The function of a honeypot is to represent itself on the internet as a potential target for attackers usually, a server or other high-value asset and to gather information and notify defenders of any attempts to access the honeypot by unauthorized users.

Honeypot systems often use hardened operating systems (OSes) where extra security measures have been taken to minimize their exposure to threats. They are usually configured so they appear to offer attackers exploitable vulnerabilities. For example, a honeypot system might appear to respond to Server Message Block (SMB) protocol requests used by the WannaCry ransomware attack and represent itself as an enterprise database server storing consumer information.

Large enterprises and companies involved in cyber security research are common users of honeypots to identify and defend against attacks from advanced persistent threat (APT) actors. Honeypots are an important tool that large organizations use to mount an active defense against attackers or for cyber security researchers who want to learn more about the tools and techniques attackers use.

The cost of maintaining a honeypot can be high, in part because of the specialized skills required to implement and administer a system that appears to expose an organization's network resources, while still preventing attackers from gaining access to any production systems.

Till now there has not been any development made in the field of intrusion and its detection. And what's more no organization has enabled itself against any kind of such intrusion as well as detection i.e.very few of them are immune against such catastrophes. The hazards of such malicious activities are numerous most prominent of them is the Data Leakage."Honeypots: Tracking Hackers" are designed to mimic systems that an intruder would like to break into but limit the intruder from having access to an entire network. If a "Honeypots: Tracking Hackers" is successful, the intruder will have no idea that s/he is being tricked and monitored. Not only this in order to ensure proper security a record of the intruder's activities is kept, so

that we; can gain insight into attack methodologies to better protect our real production systems. Proposed "Honeypots: Tracking Hackers" is a web based program to promote awareness and to stop more websites to splurge into the victims of hacking. Also give them a platform to be connected with a helping website. It should showcase support in various types of hacking, which should be motivation for preventing hacking around the globe. We are providing information throughout the world 24*7. We are providing direct interaction with the administrator. Our project provides secure access of confidential data. It becomes Easy to manage the records. Removes cost of consultation. Availability of data on a click. While developing a project, it is very important to define the category of such project. As for as this application is concern, this application can be categorized in the category of RDBMS and OOPS because this application is built to perform and deliver the primary features of RDBMS and OOPS. Various features of Database are used to maintain the database. As this application is to be built using JSP, so all the basic and primary concepts of OOPS are used. It is a web application that can be run on internet or on any other network. Its back end is RDBMS (relational database management system).

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

The term "Honeypots: Tracking Hackers" is often understood to refer as a bait which is use to lure any inter cyber malware may it be a hacker, an eavesdropper or any worm which is roaming in search of any such information (honey).

An alternative explanation for the term is a reflection of the sarcastic term for outhouses and other methods of collecting human waste in places that lack indoor plumbing. Honey being a euphemism for such waste, which is kept in a honey pot until it is picked up by a honey wagon and taken to a disposal area. In this usage, attackers are the equivalent of flies, drawn by the stench of waste.

1.1 OBJECTIVE AND SCOPE OF THE PROJECT

In the digital arms race that is **crackers vs. administrator**, tightening the existing security holes will only force the crackers to get better, while the administrator get complacent. They get the latest and greatest piece of ready-made software, and call themselves experts. What is bound to happen in the majority of the situations is that **a company will set a tracking program and never bother to spend the time it takes to maximize its effectiveness**.

Of course, the true answer is for administrator and software programmers who actually take a little pride in their work, and do their jobs properly. Also, it would help if software companies would take some responsibility when they find security holes in their product, and to update accordingly. System administrator should also feel obligated to keep their software current, and to make sure nobody within their company is given more access than they need. Hence our project "***Honeypots: Tracking Hackers***" is the sole answer to all such problems.

Proposed "***Honeypots: Tracking Hackers***" is a web based program to promote awareness and to stop more websites to splurge into the victims of hacking. Also give them a platform to be connected with a helping website. It should showcase support in various types of hacking, which should be motivation for preventing hacking around the globe.

- We are providing information throughout the world 24*7.
- We are providing direct interaction with the administrator.

1.2 MOTIVATION

Different types of honeypot can be used to identify different types of threats. Various honeypot definitions are based on the threat type that's addressed. All of them have a place in a thorough and effective cybersecurity strategy.

Email traps or spam traps place a fake email address in a hidden location where only an automated address harvester will be able to find it. Since the address isn't used for any purpose other than the spam trap, it's 100% certain that any mail coming to it is spam. All messages which contain the same content as those sent to the spam trap can be automatically blocked, and the source IP of the senders can be added to a denylist.

A **decoy database** can be set up to monitor software vulnerabilities and spot attacks exploiting insecure system architecture or using SQL injection, SQL services exploitation, or privilege abuse.

A **malware honeypot** mimics software apps and APIs to invite malware attacks. The characteristics of the malware can then be analyzed to develop anti-malware software or to close vulnerabilities in the API.

A **spider honeypot** is intended to trap webcrawlers ('spiders') by creating web pages and links only accessible to crawlers. Detecting crawlers can help we learn how to block malicious bots, as well as ad-network crawlers.

By monitoring traffic coming into the honeypot system, we can assess:

- where the cybercriminals are coming from
- the level of threat
- what modus operandi they are using
- what data or applications they are interested in
- how well our security measures are working to stop cyber attacks

1.3 EXISTING SOFTWARE

Cowrie – Cowrie is an SSH honeypot based off an earlier favourite called Kippo. It will emulate an interactive SSH server with customisable responses to commands. Another alternative is HonSHH which sits between a real SSH server and the attacker, MiTMing the connection and logging all SSH communications.

Dionaea is a multi-protocol honeypot that covers everything from FTP to SIP (VoIP attacks). Where it really excels is for SMB decoys. It can even simulate malware payload execution using LibEmu to analyse multi-part stagers.

Honeything emulates the TR-069 WAN management protocol, as well as a RomPager web-server, with vulnerabilities. Other IoT decoys can be created by emulating embedded telnet / FTP servers, for example with BusyBox.

ConPot emulates a number of operational technology control systems infrastructure. These include protocols like MODBUS, DNP3 and BACNET. It comes with a web-server that can emulate a SCADA HMI as well.

GasPot emulates a Veeder Root Gaurdian AST that is commonly used for monitoring in the oil and gas industry.

MongoDB-HoneyProxy emulates an insecure MongoDB database. Hackers regularly scan the interwebs looking for administrators who had an ‘oops moment’ and exposed their DB to the world.

ElasticHoney emulates an ElasticSearch instance, and looks for attempted remote code execution.

1.4 PROBLEM DEFINITION

Till now there hasn't been any development made in the field of **intrusion and its detection**. And what's more no organization has enabled itself against any kind of such intrusion as well as detection i.e. very few of them are immune against such catastrophes. The hazards of such malicious activities are numerous most prominent of them is the **Data Leakage.**

*"**Honeypots: Tracking Hackers**"* are designed to mimic systems that an intruder would like to break into but **limit the intruder from having access to an entire network**. If a "Honeypots: Tracking Hackers" is successful, the intruder will have no idea that s/he is being tricked and monitored.

Not only this in order to ensure proper security **a record of the intruder's activities is kept, so that we; can gain insight into attack methodologies to better protect our real production systems.**

Honey pots are usually programs that emulate services on a designated port, but once successfully cracked, offer no real power to the attacker. The "**Honeypots: Tracking Hackers**" program will then alert the admin that an attack is in progress, and will allow the admin to track the attacker's every move. Honeypots will also show the methods the attacker is using to gain entry, and what methods the attacker is using to cover his or her tracks.

While it is often a computer, a "**Honeypots: Tracking Hackers**" can take other forms, such as files or data records, or even unused **IP address** space. A honey pot that **masquerades as an open proxy** to monitor and record those using the system as bait.

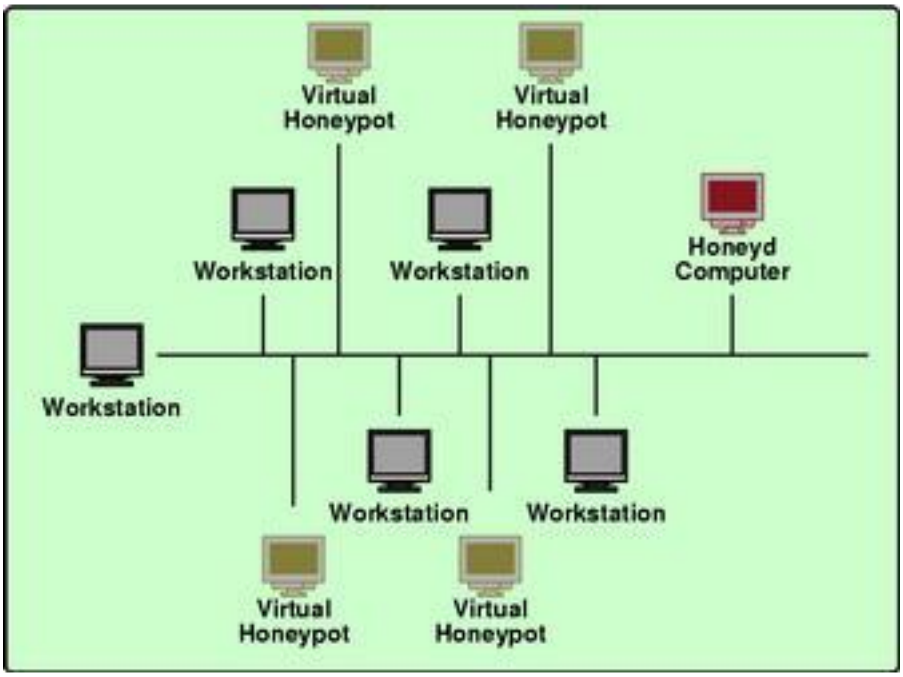

Figure 1.1 Block diagram of Honeypot System

CHAPTER 2

2.1 BACKGROUND AND RELATED WORK

Most of the systems in use today, therefore, work with a set of so-called ***attack signatures***, that describe **attacking patterns** in sufficient detail for identifying ongoing attacks automatically.

However, the specification of such signatures usually needs to be done by experienced network

security analysts by either monitoring an existing network and extracting the relevant information, this is the **major loop hole in the whole process** and using this glitch any outsider can or may gain access to company's relevant data.

Table. 2.1 Comparative Study of various honeypots systems

Sr.No.	Paper Name	Author	Publication Year	Methodology
1	Honeypot for Cyber security Threat Intelligence	Ren Rui Tan, Simon Eng, Kiam Cheng How, Yongqing Zhu and Paul Wu Horng Jyh	2022	SME, Cybersecurity Threat Intelligence, Web Honeypot; Threat Intelligence Dashboard First Section
2	Honeypot Software and Data Analysis	Marcin Nawrocki, Matthias Wahlisch, Thomas C. Schmidt	2016	Extensive overview on honeypots. This includes not only honeypot software but also methodologies to analyse honeypot data.
3	Recent Advances and Future Trends	Matthew L. Bringer, Christopher A.	2015	Honeypots, Alarm system, Computer

	in Honeypot Research	Chelmecki, and Hiroshi Fujinoki		hacking, Computer crime, Computer security
4	Honeypot in Network Security: A Survey	Abhishek Mairh, Debabrat Barik, Kanchan Verma, Debasish Jena	2013	Honeypot, Honeycomb, Honeynet, IDS, Load balancer and Honeywall.
5	Honeypot: Tracking Hackers	Lance Spitzner, Addison-Wesley	2018	Tracking Hackers methods
6	Honeypots: Approach and Implementation	Kumar Shridhar, Mayank Jain	2013	Network attack, N-Map, DMZ, Network Security
7	Honeypot: A Trap for Attackers	Savita Paliwal	2017	Low level interaction honeypot, medium level interaction honeypot, high level interaction honeypot and functions of honeypots.
8	A survey of honeypot research: Trends and opportunities	Ronald M. Campbell, Keshnee Padayachee, Themba Masombuka	2015	Devices connected to computer networks and its threat

CHAPTER 3

HARDWARE AND SOFTWARE REQUIREMENT

For software development very first and essential requirement is the availability of software and hardware. The software requires the following software and hardware for its successful implementation.

3.1 Software Requirement

3.1.1 Client Side (Recommended)

Any Operating System with Web Browser.
Internet or LAN connection

3.1.2 Server Side (Recommended)

Operating System	:	WINDOWS XP /98/2000/WIN 7/LINUX OR ANY OS
Scripting Language	:	JAVA SCRIPT
Front-End Language	:	JSP AND JDK1.6
Back-End	:	SQL-SERVER 2000
Mark-up Language	:	HTML4.0, DHTML
Web-Server	:	WEB-LOGIC or TOMCAT

3.2 Hardware Requirement

3.2.1 Client Side (Recommended)

Processor	:	Pentium-4 or above
RAM	:	1GB
Hard Disk	:	160GB
Keyboard	:	Any
Mouse	:	Any

3.2.2 Server Side (Minimum)

Processor	:	Intel-C2D
RAM	:	4 GB
Hard Disk	:	500 GB
Keyboard	:	Any
Mouse	:	Any

CHAPTER 4

SDLC METHODOLOGIES

SDLC stands for Software Development Life Cycle models, and these are a variety of processes of design, development and testing that are used in the industry today. While there's no best or standout SDLC methodology, it's essential to be across the most common models that can be applied to projects within a company.

4.1.1 AGILE METHODOLOGY

Agile is a combination of an incremental and iterative approach, where the product is released on an ongoing cycle then tested and improved at each iteration. Fast failure is encouraged in agile methodology: the theory is that if we fail fast and early, we can solve minor issues before they grow into major issues.

Agile is one of the most common methodologies out there today but it's technically more of a framework than a distinct model. Within Agile, there are sub-models in place such as extreme programming (XP), Rapid Application Development (RAD), Kanban and Scrum methodology.

4.1.2 WATERFALL METHODOLOGY

The Waterfall methodology is one of the oldest surviving SDLC methodologies. It follows a straightforward approach: the project development team completes one phase at a time, and each phase uses information from the last one to move forward.

While this methodology does make the needs and outcomes clear, and gives each stage of the model a well-defined starting and ending point, there are downsides in Waterfall's rigidity. In fact, some experts believe the Waterfall model was never meant to be a working SDLC methodology for developing software because of how fixed it is in nature. Because of this, SDLC Waterfall methods are best used for extremely predictable projects.

4.1.3 DEV-OPS

DevOps is used by some of the biggest companies out there, such as Atlassian. A hybrid of Agile and Lean, DevOps evolved from the growing need for collaboration between operations and development teams throughout the SDLC process. In DevOps, both developers and operations teams work together to accelerate and innovate the deployment and creation of software. There are small but frequent updates and DevOps encourages continuous feedback, process improvement and the automation of previously manual processes.

DevOps methodology saves time and improves communication because both operations and development teams get to know about the potential obstructions at the same time. However, DevOps may open software to more security issues, as this approach generally favors speed over security.

4.1.4 SPIRAL

As one of the most flexible SDLC models out there, the Spiral model is used by the world’s leading software companies. Spiral enables project teams to build a highly customized product. Spiral methodology passes through four phases repeatedly until the project is finished: planning, risk analysis, engineering, and evaluation.

The biggest difference between Spiral and other methodologies is that it is focused on risk analysis, with each iteration it focuses on mitigating potential risks. The model also emphasizes customer feedback, and as the prototype build is done in small increments, cost estimation becomes easier.

4.1.5 ITERATIVE

The Iterative model is all about repetition. Instead of starting out with a comprehensive overview of the requirements, development teams build the software piece by piece and identify further requirements as they go along. As a result, each new phase in the Iterative model produces a newer, more-refined version of the software under development.

Iterative allows developers and testers to identify functional or design flaws early, and can easily adapt to the ever-changing needs of the client. Like Spiral, Iterative suits larger projects and requires more management and oversight to work well.

4.2 HOW IT AFFECT OUR MODEL

Honeypot is a complete package of all hacker tracking components. A honey pot is a trap set to detect, deflect, or in some manner counteract attempts at unauthorized use of information systems. Generally it consists of a computer, data, or a network site that appears to be part of a network, but is actually isolated, (un) protected, and monitored, and which seems to contain information or a resource of value to attackers. Iterative model applies to our project as we created feedback models. We created small models first such as admin module, login/sign-up module, customer module, etc and then tested separately. From those feed backs, we resolved issues and then create new models. The basic idea behind this method is to develop a system through repeated cycles (iterative) and in smaller portions at a time (incremental).

CHAPTER 5

ER/ ACTIVITY/ USECASE DIAGRAMS

5.1 USE CASE DIAGRAM

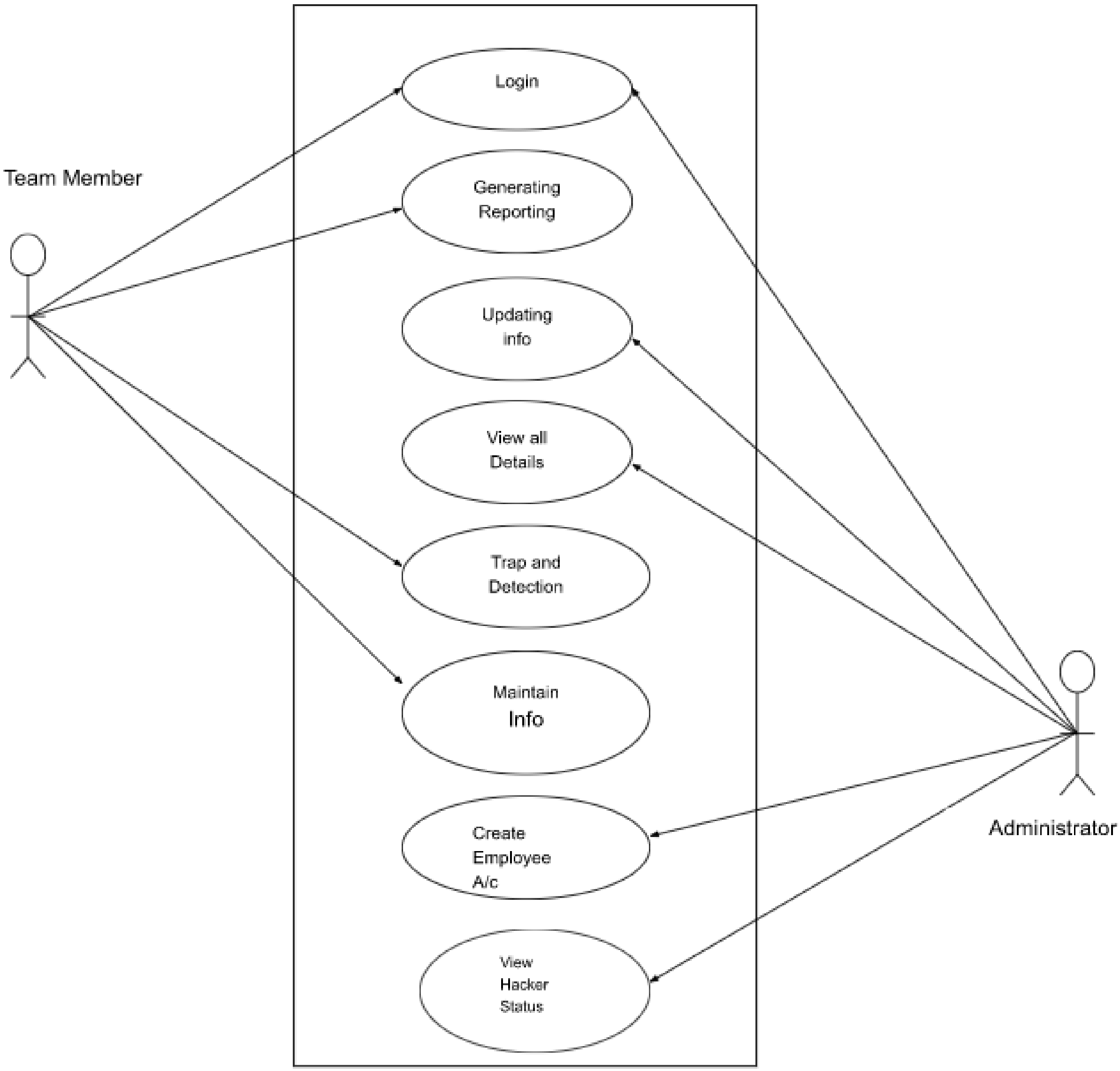

Figure 5.1 Use Case Diagram

5.2 ACTIVITY DIAGRAMS

5.2.1 LOGIN MODULE

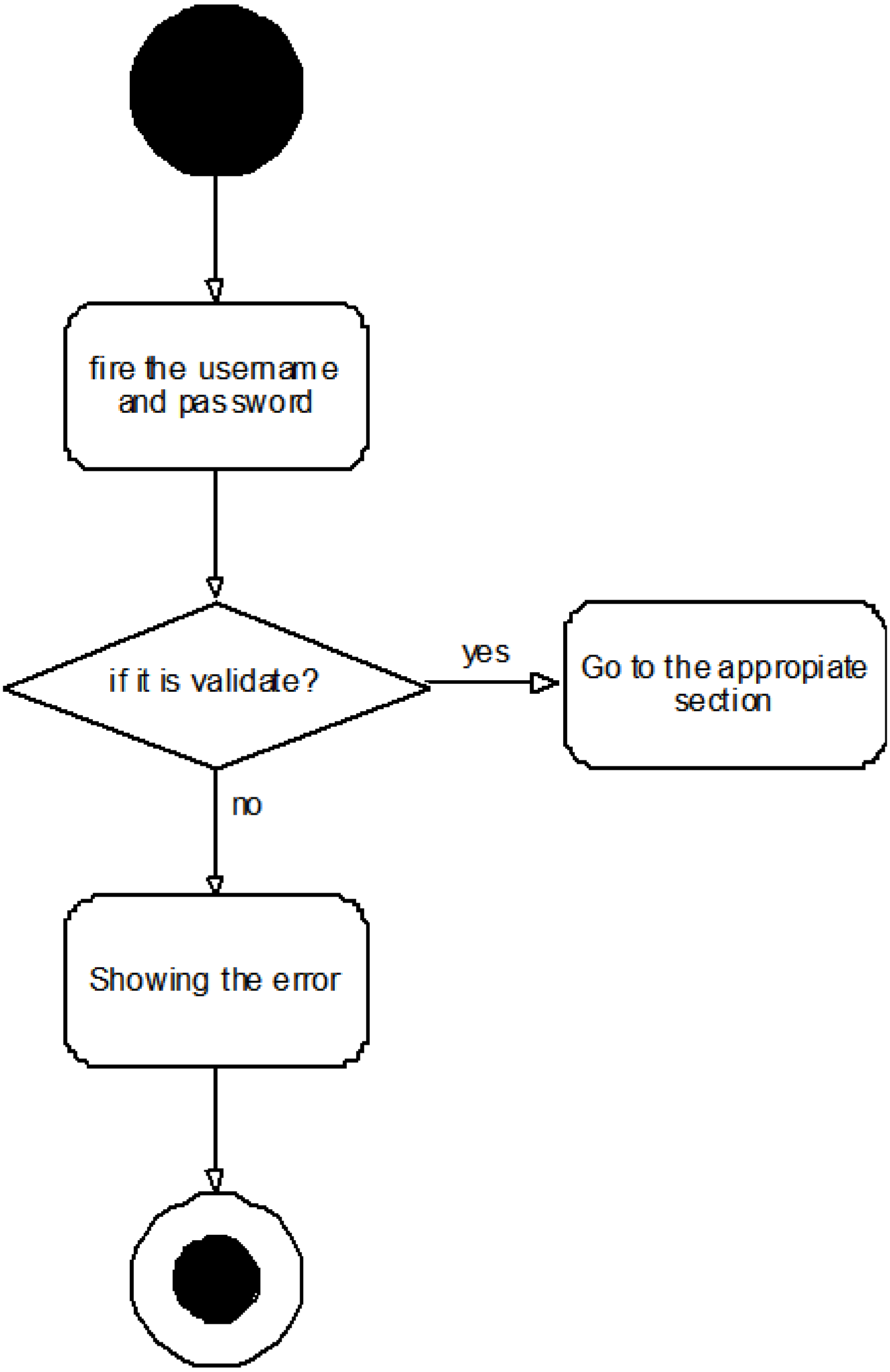

Figure 5.2 Login Module

5.2.2 ADMIN MODULE

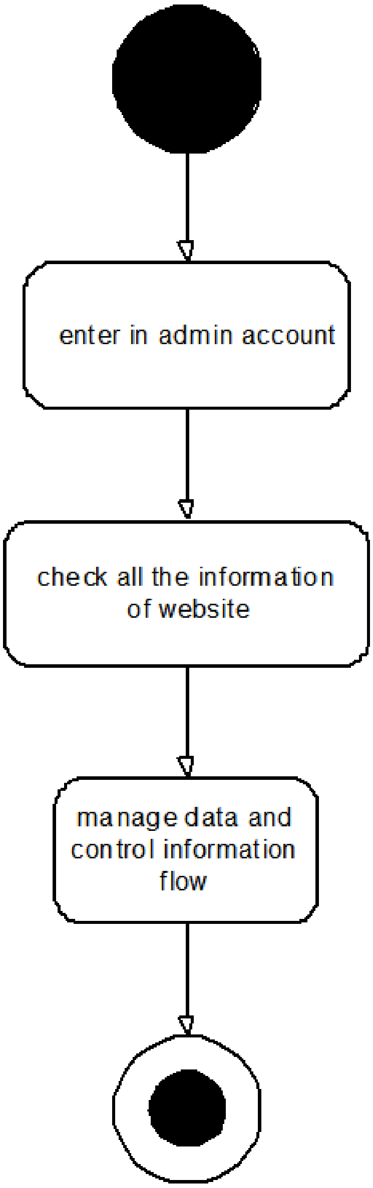
enter in admin account
check all the information of website
manage data and control information flow

Figure 5.3 Admin Module

5.2.3 HACKER MODULE

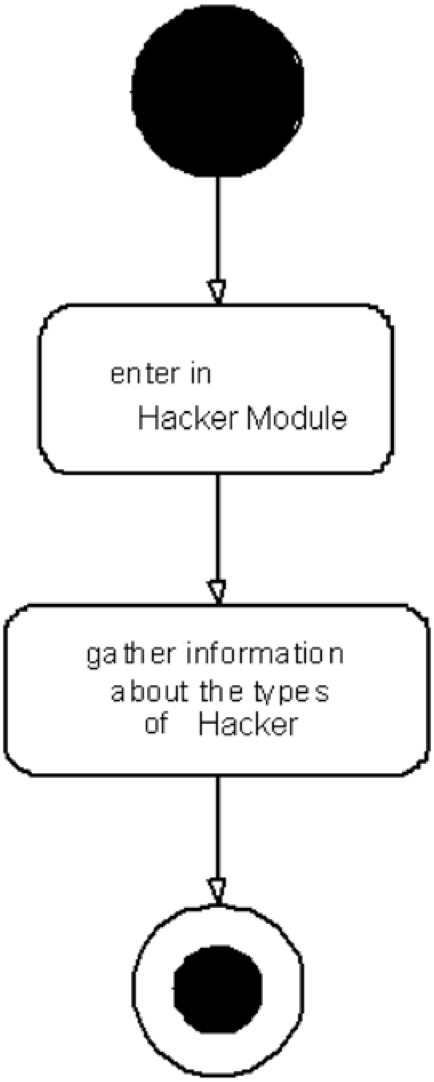

Figure 5.4 Hacker Module

5.2.4 CUSTOMER MODULE

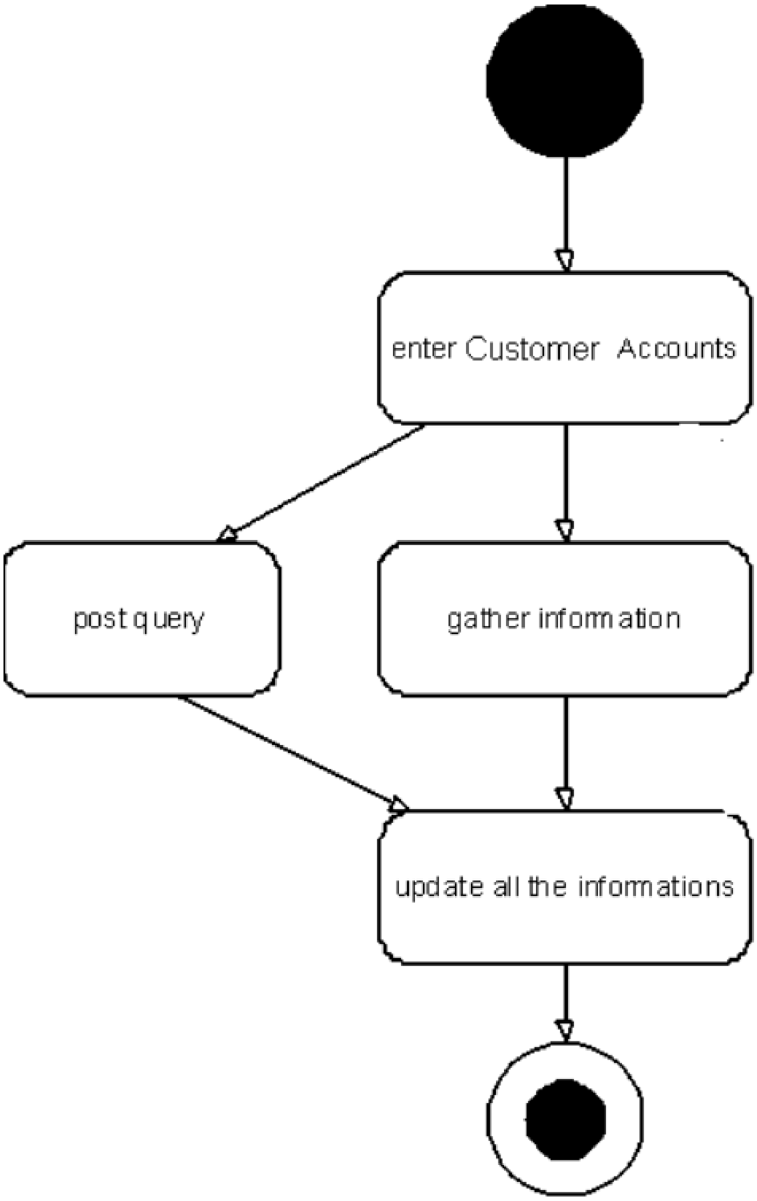

Figure 5.5 Customer Module

5.3 ER DIAGRAM

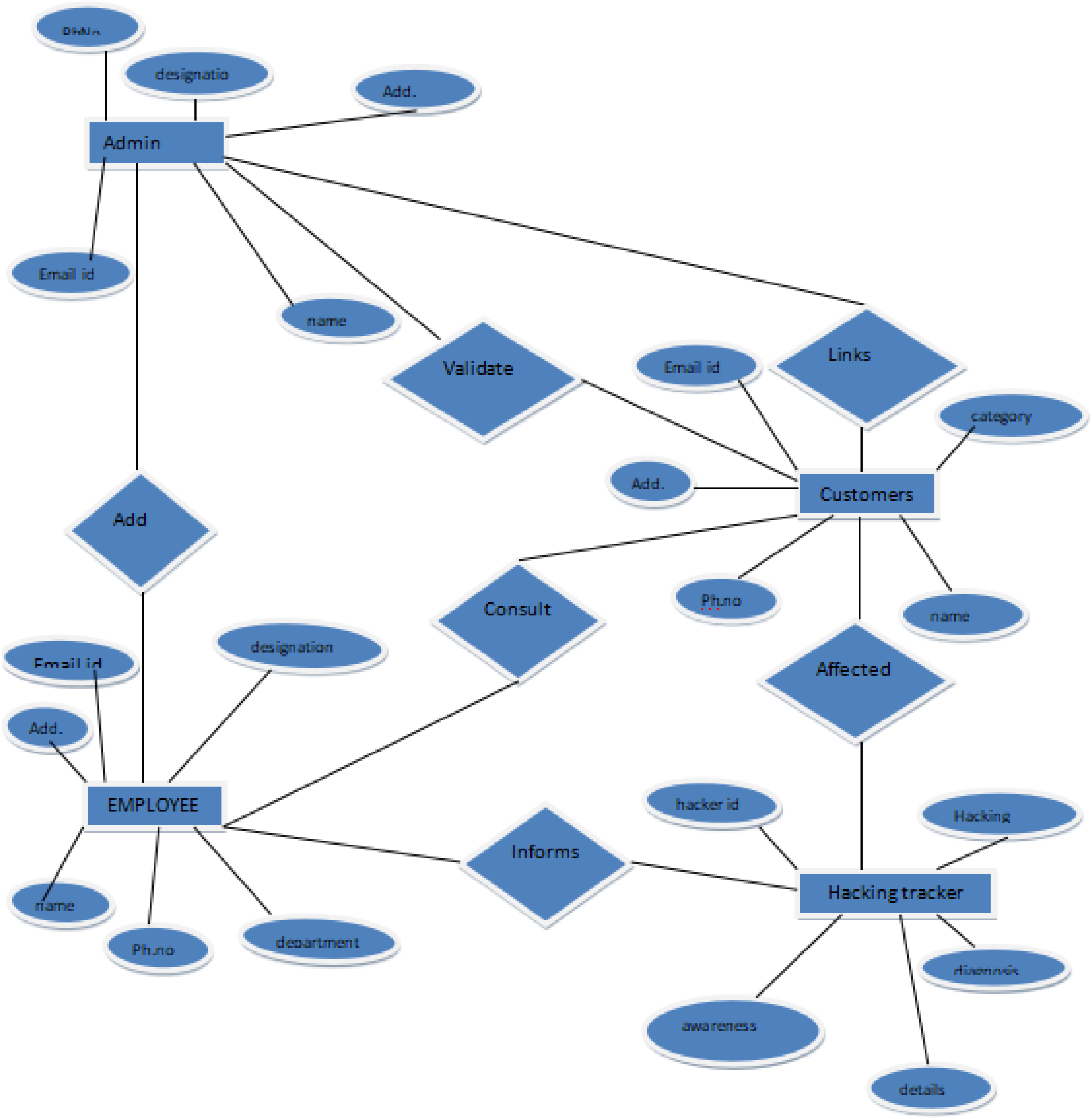

Figure 5.6 ER Model for Honeypots: Tracking Hackers

BIBLIOGRAPHY

[1] Honeypot for Cybersecurity Threat Intelligence
https://www.researchgate.net/publication/365300476_Honeypot_for_Cybersecurity_Threat_Intelligence

[2] JavaScript by Wrox Publication

https://p2p.wrox.com/javascript-446/index.html

[3] Honeypot: Tracking Hackers

Lance Spitzner, Addison-Wesley Professional, Software Architecture in Practice, 2nd ed. Reading, MA: Addison Wesley, 2002. [E-book] Available: https://www.oreilly.com/library/view/honeypots-tracking-hackers/0321108957/

[4] Honeypot in Network Security: A Survey

V. Paxson, .Bro: A System for Detecting Network Intruders in Real-Time. Computer Networks (Amsterdam, Netherlands: 1999), vol. 31, no. 23-24, pp. 2435.2463, 1998. Available:https://www.researchgate.net/publication/220846415_Honeypot_in_network_security_A_survey

[5] Honeypots: Approach and Implementation

International Journal of Science and Research (IJSR)

https://www.ijsr.net/

[6] The top 7 SDLC methodologies

https://www.michaelpage.com.au/advice/career-advice/productivity-and-performance/top-7-sdlc-methodologies