Configuration Manual

MSc Research Project

MSc Cybersecurity

Iqra Fareed

Student ID: x23163461

School of Computing

National College of Ireland

Supervisor: Vikas Sahni

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: ……. …**Iqra Fareed**……………………………………………………………………..…….…

Student ID: …………**x23163461**…………………………………………………………………..……..

Programme: ………**MSc Cybersecurity**…….………………… **Year:** ….**2024**……….……..

Module: ……………………..**Practicum**……………………………………………………………….………

Lecturer: …………………**Vikas Sahni**…………………………………………………….……..………

Submission Due Date: ………………**16/09/2024**….……………………………………………………………..………

Project Title: **A Game Theoretic and Machine Learning Approach for Strengthening Network Security Using Honeypots in Healthcare**

Word Count: ……………**1684**…………… **Page Count:** ……………….…..**8**……………….……..…..……

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Iqrafareed ……………………………………………………………………………………………..….

Date: …………………………………**16/09/2024**…….…………………………………………………

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual

Iqra Fareed
X23163461

1 Introduction

This configuration manual is designed to give users and the reader of this research comprehensive instructions on the setup and administration of honeypots to enhance network protection. The intended audience of this manual is the people who want to use honeypots as a tool for detecting and studying threats while not endangering their infrastructure. A honeypot is a developed security concept aimed at attracting potential attackers to a fake target instead of the actual target. Such a decoy system is helpful to network administrators since it allows them to monitor the various unlawful activities being conducted within the network and understand the techniques the attackers are using.

2 System Requirements

2.1 Software dependencies

Table 2.1: System Specifications

Requirement	Minimum Specification
Network	Dedicated IP Address
Network Segmentation	Isolated VLAN
Software Dependencies	Python 3.12.3

Software Dependencies: The honeypot depends on several software, as stated below. The honeypot is coded in Python-3.12.3 [2], and the coding environment used is Google Colab [1]. Python is used to execute the scripts and automate the processes. The remaining dependencies are Python libraries and modules used for machine learning model training, dataset cleaning and preprocessing, and model exportation. Moreover, implementing the Bayesian game theory model also requires certain libraries to be pre-installed and updated to the latest available version. These dependencies need to be fulfilled to run and deploy this on a new system; otherwise, the system won't work.

2.2 Design specifications

1. **Network:** The honeypot concept is based on the idea that it should be easy to monitor and collect data on, thus the relevance of a dedicated IP (Internet Protocol) address. This

[1] https://colab.research.google.com/
[2] https://www.python.org/downloads/release/python-3123/
[3] https://www.kaggle.com/datasets/subhajournal/android-ransomware-detection

separation assists in monitoring all the activities with the honeypot independently of other communications in the network.

2. **Network Segmentation:** Having the honeypot in a separate VLAN (Virtual Local Area Network) is highly effective and important regarding security. This setup ensures that individuals cannot transverse to other parts of the network if they gain access to the honeypot. Administrative isolation means that a honeypot is a separate network, and such risks are minimised in the critical networks.
3. **Dataset:** The Android ransomware dataset obtained from Kaggle is used for model training [3].

2.3 Libraries

2.3.1 Bayesian

Library	Function	Version
scipy.stats.beta	It provides functions related to the beta distribution and is commonly used in Bayesian statistics.	1.6.2
nashpy	They are used to create and analyse 2-player games, specifically for finding Nash equilibrium in game theory.	0.0.41

2.3.2 Model training

Library	Function	Version
scikit-learn.preprocessing.LabelEncoder	Encodes categorical variables into numeric labels, transforming non-numeric labels (as long as they are hashable and comparable) to integers.	1.3.0
scikit-learn.preprocessing.StandardScaler	Standardises features by removing the mean and scaling to unit variance, preparing data for machine learning algorithms.	1.3.0
scikit-learn.preprocessing.MinMaxScaler	Transforms features by scaling each feature to a given range, typically between zero and one.	1.3.0

[1] https://colab.research.google.com/
[2] https://www.python.org/downloads/release/python-3123/
[3] https://www.kaggle.com/datasets/subhajournal/android-ransomware-detection

2.3.3 Supporting libraries

Library	Function	Version
Pandas	Used for data manipulation and analysis, providing data structures like DataFrames.	2.1.1
NumPy	Provides support for large, multi-dimensional arrays, matrices, and mathematical functions.	1.26.0
Matplotlib.pyplot	A plotting library used for creating static, animated, and interactive visualisations in Python.	3.8.0
Seaborn	Built on Matplotlib, it makes statistical graphics, especially for data visualisation.	0.13.0
math	Provides mathematical functions like square root, trigonometric functions, and logarithms.	Built-In
joblib	It provides utilities for saving and loading Python objects to disk, and it is often used to save trained models.	1.3.2
networkx	A library for creating, manipulating, and studying complex networks' structure, dynamics, and functions.	3.1

2.4 Modules

2.4.1 Model Training Modules

Module	Function	Version
scikit-learn.model_selection.train_test_split	Splits data into training and testing sets.	1.3.0
scikit-learn.linear_model.LogisticRegression	Implements the Logistic Regression algorithm for classification tasks.	1.3.0
scikit-learn.tree.DecisionTreeClassifier	Implements the Decision Tree algorithm for classification tasks.	1.3.0
scikit-	Implements the Random	1.3.0

learn.ensemble.RandomForestClassifier	Forest algorithm for classification, an ensemble method based on decision trees.	
scikit-learn.neighbors.KNeighborsClassifier	Implements the K-Nearest Neighbors algorithm for classification tasks.	1.3.0
scikit-learn.ensemble.GradientBoostingClassifier	Implements the Gradient Boosting algorithm for classification, another ensemble method.	1.3.0

2.4.2 Evaluation Metrics Modules

Module	Function	Version
scikit-learn.metrics.accuracy_score	Computes the accuracy of the classification model.	1.3.0
scikit-learn.metrics.precision_score	Computes the precision of the classification model, which is the ratio of correctly predicted positive observations to the total predicted positives.	1.3.0
scikit-learn.metrics.recall_score	Computes the recall of the classification model, which is the ratio of correctly predicted positive observations to all observations in the actual class.	1.3.0
scikit-learn.metrics.f1_score	Computes the F1 score, the weighted average of precision and recall.	1.3.0
Scikit-learn.metrics.classification_report	Generates a detailed classification report, including precision, recall, and F1-score for each class.	1.3.0

2.4.3 Supporting Visualisation/Plotting Libraries

Module	Function	Version
scikitplot.metrics	Provides additional plotting capabilities for evaluating model performance (e.g., confusion matrices, ROC curves).	0.3.7

2.5 Network Requirements

2.5.1 Network architecture

Network design plays a crucial role in honeypots; hence, an efficient network design is essential for the integration of honeypots. The primary location of resources is the healthcare's central or core network, which may comprise a data centre alongside other applications, databases, and servers. These are the EHR systems, application servers for telecommunication and patient portals, and DB (Database) servers for patients and other organisational data. The access layer incorporates the end user's true pocket PCs, which include workstations, mobile devices, and medical IoT devices used by caretakers to interface with the network. Security services are deployed as additional layers to the network, and as part of this strategy, there are firewalls, IDS/IPS, and VPN gateways. Technical administration and support are also vital, including the environment's backup and recovery mechanisms, monitoring tools for the network, and Identity and Access Management systems. To obtain an optimal security perspective, honeypots are implemented in this structure; their objective is to identify and analyse threatening actions.

2.5.2 Network settings

Setting up and configuring networks play an essential role in implementing honeypots. To make the monitoring and management of the honeypots easier and more consistent, static IP addresses should be used. Honeypots should be put in different subnets or VLANs as they are important in minimising the lateral physical movement of the attackers. This segmentation assists in managing potential threats and reduces the impact on the critical system. Also, the DNS settings should be tuned to allow honeypots to resolve required domain names for mostly logging and alerting. Segmenting the network appropriately and isolating honeypots is extremely important so you won't have an environment that is too out of control.

2.5.3 Network configuration

Some control measures to enhance the creation of honeypots are as follows: Proper network configurations such as firewalls, IDS/IPS and VPN gateways should be implemented to protect the honeypots. Firewalls should then be set to allow all traffic to and from the honeypot and deny all traffic. This ensures that only the right and genuine communication is provided and stored in the database. IDS/IPS systems must be deployed to capture pal traffic going in and coming out of the honeypot and analyse the traffic in parallel in real-time for odd behaviour. To prevent the introduction of new threats by management activities, VPN gateways should be configured to allow secure, encrypted, remote access only to the honeypot administrators. Both configurations, in combination, improve the security and efficiency of the honeypot implementation (NIST, 2020).

2.5.4 Devices for Deployment

There are a few gadgets that are crucial to the running and managing of honeypots. Workspaces in nurse stations and doctors' offices are the main touchpoints for accessing patients' information and other applications in the healthcare organisation. The healthcare staff utilise portable end-user devices, such as tablet computers and smartphones, to retrieve information on the move. IoT (Internet of Things) of patients' uses, such as the heart monitor and infusion pump, deliver actual-time details and can communicate via the network. Network firewalls are defined as being situated at the external perimeters of the networks and can be either a part of the hardware or software type. IDS/IPS (Intrusion Detection/Prevention Systems) devices are further configured to enable intrusion detection and prevention; VPN (Virtual Private Network) gateways, on the other hand, are used to grant distant user access securely. Honeypot effectiveness largely depends on the devices to be targeted; this requires the proper devices to be implemented (NIST, 2019).

References

National Institute of Standards and Technology (NIST) (2019) Guidelines for the Selection, Configuration, and Use of Transport Layer Security (TLS) Implementations. NIST Special Publication 800-52 Revision 2. Available at: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-52r2.pdf (Accessed: 7 August 2024).

National Institute of Standards and Technology (NIST) (2020) Security and Privacy Controls for Information Systems and Organizations. NIST Special Publication 800-53 Revision 5. Available at: https://csrc.nist.gov/publications/detail/sp/800-53/rev-5/final (Accessed: 7 August 2024).