Combating Crypto-jacking through machine learning

Project Team

Istafa Malik	19P-0033
Muhammad Affan Khan	19P-0045
Rana Rehan Qaisar	19P-0077

Session 2019-2023

Supervised by

Dr. Muhammad Amin

Department of Computer Science

National University of Computer and Emerging Sciences
Peshawar, Pakistan

June, 2022

Student's Declaration

We declare that this project titled "*Combating Crypto-jacking through machine learning*", submitted as requirement for the award of degree of Bachelors in Computer Science, does not contain any material previously submitted for a degree in any university; and that to the best of our knowledge, it does not contain any materials previously published or written by another person except where due reference is made in the text.

We understand that the management of Department of Computer Science, National University of Computer and Emerging Sciences, has a zero tolerance policy towards plagiarism. Therefore, We, as authors of the above-mentioned thesis, solemnly declare that no portion of our thesis has been plagiarized and any material used in the thesis from other sources is properly referenced.

We further understand that if we are found guilty of any form of plagiarism in the thesis work even after graduation, the University reserves the right to revoke our BS degree.

Istafa Malik Signature: ________________

Muhammad Affan Khan Signature: ________________

Rana Rehan Qaisar Signature: ________________

____

Verified by Plagiarism Cell Officer
Dated:

Certificate of Approval

The Department of Computer Science, National University of Computer and Emerging Sciences, accepts this thesis titled *Combating Crypto-jacking through machine learning*, submitted by Istafa Malik (19P-0033), Muhammad Affan Khan (19P-0045), and Rana Rehan Qaisar (19P-0077), in its current form, and it is satisfying the dissertation requirements for the award of Bachelors Degree in Computer Science.

Supervisor

Dr. Muhammad Amin Signature: ____________________

Coordinator of FYP

FYP Coordinator
National University of Computer and Emerging Sciences, Peshawar

Dr. Head of Computer Science Department

HoD of Department of Computer Science
National University of Computer and Emerging Sciences

Acknowledgements

Your acknowledgments here

Istafa Malik
Muhammad Affan Khan
Rana Rehan Qaisar

Abstract

Combating cryptojacking based on network traffic analysis represent an active area of research and various techniques have been presented over the years with varying degrees of success. One of the recent and Latest technologies is through network traffic analysis which monitors the traffic and then analyses the change in traffic after the cryptojacking is done.By preventing cryptojacking activities, computer and device users can enjoy better performance from their devices. Cryptojacking can significantly slow down the performance of a device, which can impact the user's productivity.The methodology used is machine learning-based detection.This methodology involves training machine learning models to identify patterns in network traffic that are indicative of cryptojacking.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In today's digital environment, the excessive use of cryptocurrencies has brought new cyber security threats. One of those threats is cryptojacking, which involves the unauthorized use of computing resources to mine crypto currencies without the owner's knowledge or consent. Such encryption incidents has created an urgent need for detection and prevention mechanisms that can protect users from theft. Machine learning algorithms offers promising solutions to combat crypto-jacking. Using artificial intelligence and data analytics, machine learning can provide automatic and proactive protection against this evolving threat. Machine learning algorithms have the ability to analyze system behavior, network traffic, and other related data to identify patterns, anomalies associated with cryptographic activity and learn from the data to generate a solution. The major advantage of using machine learning algorithms to detect crypto-jacking is its ability to adapt and learn from new attack techniques. When crypto hunters use sophisticated and complex evasion tactics, traditional rules-based systems often struggle to keep up. Machine learning algorithms can continuously train and update their models to detect new patterns and signatures, improving detection accuracy and reducing false positives. Machine learning can also enable real-time monitoring and immediate response to encryption attempts. Using algorithms that can quickly process and analyze large amounts of data, potential threats can be identified, minimizing the impact on system performance, power consumption and security. However, developing an effective and precise machine learning-based solution to combat crypto-jacking is challenging. This requires labeled datasets for training, computational resources to handle complex models, and the ability to interpret, understand and explain the outputs made by these algorithms. In addition, privacy protections and regulatory compliance must be carefully reviewed to ensure ethical and responsible use of collected data. Despite all these challenges, the potential benefits of using machine learning algorithms to combat crypto-jacking are enormous. By proactively detecting and preventing unauthorized encryption, individuals and organizations can protect their computing resources, their crypto currencies, maintain system performance, and protect private data from hazards. In this project, we are investigating the application of machine learning algorithms to combat crypto-jacking. By developing simple and adaptable models, we plan to develop a defense mechanism that stays ahead of evolving crypto-jacking technologies. Through our research, data analysis and collaboration with cyber security experts, we aim to develop a solution that improves the security and resilience of digital systems in the face of this emerging threat.

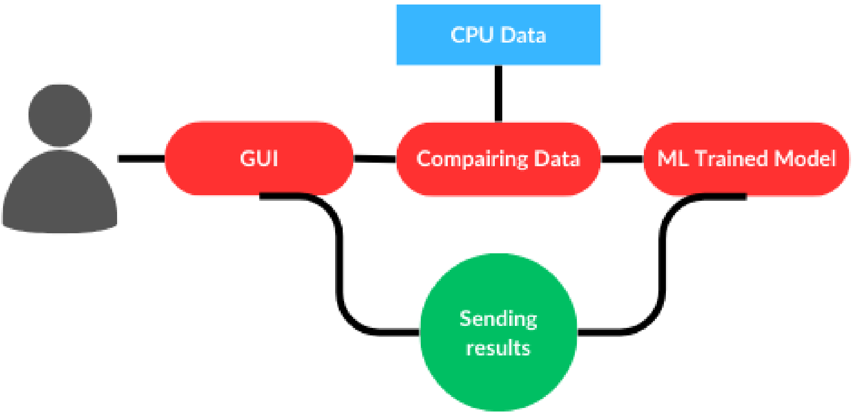


Figure 1.1: Predicted System

Chapter 2

Review of Literature

2.0.1 Paper1

The work put forward a solution that was used for prevention of possible two cyber threats. Descriptive and informative methods were not frequently used for the discussion of the origin of the file malware and cryptographic attacks.[1] As a result, all files are deducted in size Comparatively speaking, browser cryptic search has received more research than crypto-jacking. As a result, there were limitations on visibility and some challenges with determination.[3]

2.0.2 Paper2

The solution mentioned in the paper used a hybrid model that combined rule-based and machine learning algorithms for prevention. As a result, it could detected outdated data patterns but could not detect any unique or new patterns.[2] Due to the limited availability of data, there are rule-based detection limits which also limits our defense.

2.0.3 Paper3

The paper proposed a solution that used network traffic analysis approach to check and identify the crpto activities,it analysed the traffic of network to identify the activities.[4]

The system used models such as random forest, k-fold cross validation techniques to detect any malware or theft.

2.0.4 Paper4

This paper proposed a solution using hybrid strategies to avoid and postpone the execution.This system consisted of two different approaches.Firstly a model-based and hidden browser,Secondly the based mining assault models known as Delay CJ.This system consists of methodologies that are covert browser based mining attack model named Delay CJ and Model based. The conclusion was that four methods failed to detect CJ delay, validating the attack. Limitations are CJ detector needs port access and Chrome Devtools Protocol for function info, which can be impractical.[2]

Chapter 3

Project Vision

3.1 Problem Statement

Attackers are continually changing their strategies to avoid detection,current methods employ outdated data. In order to prevent crypto jacking, a strong strategy utilising machine learning techniques is needed

3.2 Business Opportunity

One can charge customers based on the number of protected devices or offer subscription packages.Use of machine learning gives us many benefits for the project, such as greater accuracy in identifying new and evolving cryptographic detection techniques and the ability to adapt to changing threats.

3.3 Objectives

Main objective is to create a system that uses machine learning to recognize crypto jacking behavior.To investigate the use of machine learning techniques to predict the ongoing crypto jacking attempts using past data.To make the machine learning-based system's detection and reporting of crypto jacking behavior easier, provide it a user-friendly interface.To evaluate the performance and dependability of the machine learning-based system, run it through real-world scenarios.

3.4 Project Scope

The cryptojacking attacks are increasing in the ratio day by day which is why the old methods are not effecient. By using machine learning algorithms to analyze the system's performance, it is possible to detect cryptojacking attack in real time.Large amounts of data can be accurately and efficiently analyzed by machine learning algorithms. They are also able to assess the performance measures and trends that are difficult for human beings to detect.

3.5 Constraints

Availability of data set is very rare and limited. Collecting and maintaining a diverse data set to train and validate machine learning models requires significant effort and resources. It's vital to reach a balance between detection accuracy to reduce false positives, which identify benign activities, and false negatives, which identify actual cryptocurrency transactions.. Careful configuration and performance assessment of the model are essential to get the ideal balance. The costs of gathering data and training models are incurred during the development and upkeep of a machine learning system.

Chapter 4

Software Requirements Specifications

4.1 Use Cases/ Use Case Diagram

Following is the use case diagram of the usecase:

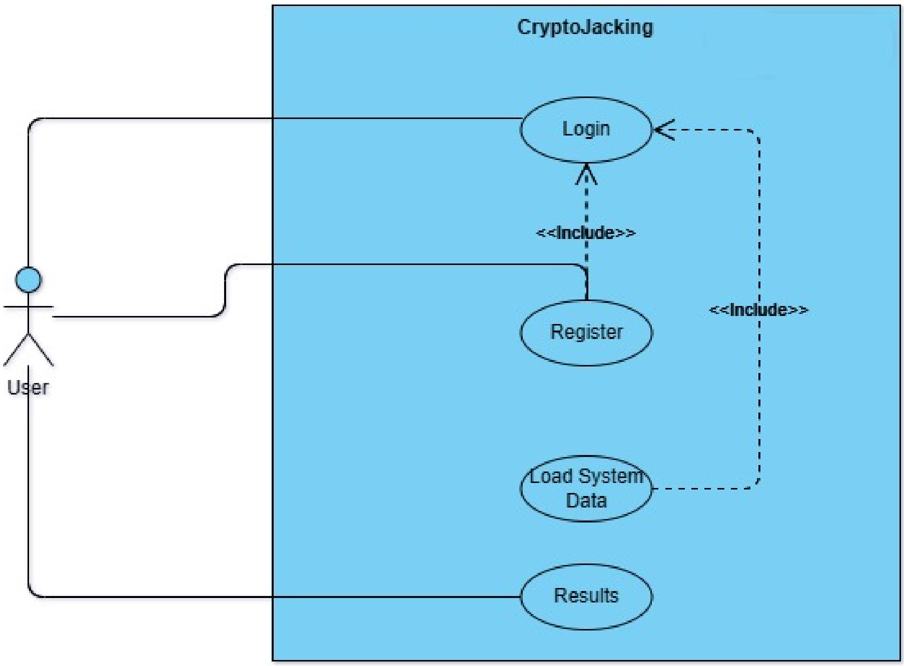


Figure 4.1: Use case diagram

4.2 Swimlane Diagram

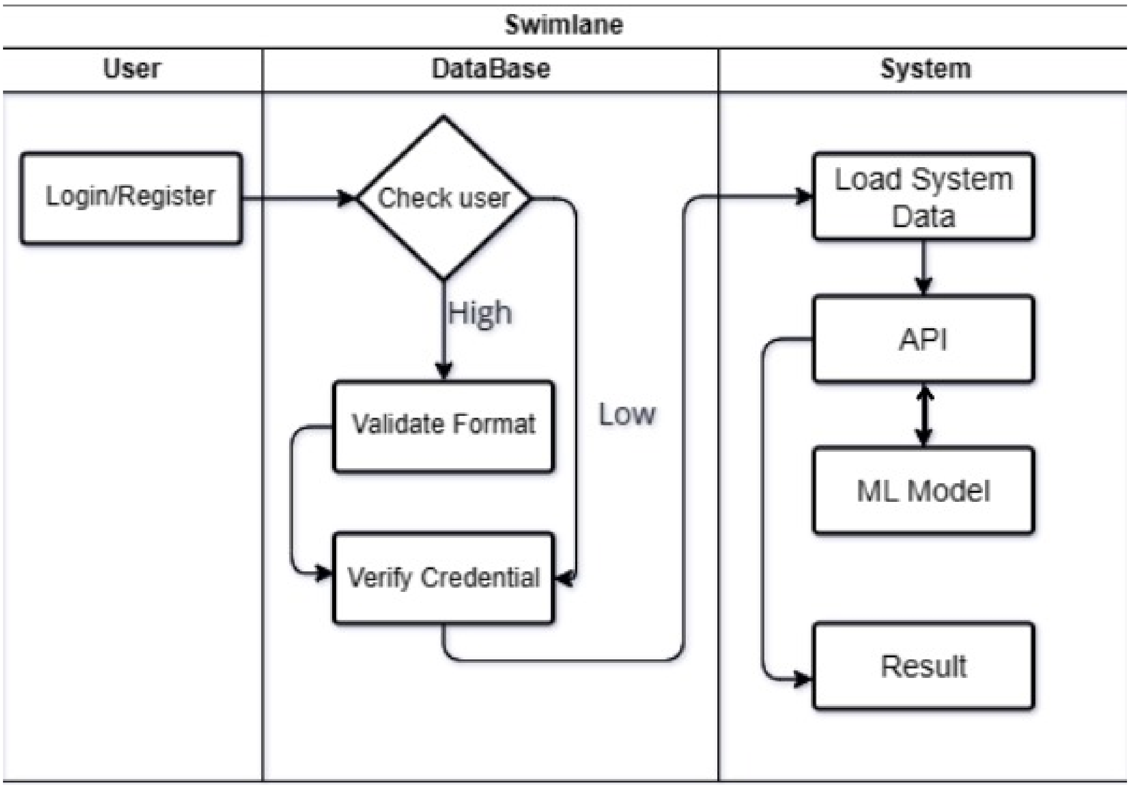


Figure 4.2: Swimlane diagram

Chapter 5

Iteration Plan

- Midterm FYP 1

  The first iteration plan is to gather data set.After the gathering of data set, prepossessing the data set is the next step in which we remove the extra columns,Null values and many other to improve the accuracy of our final product.

- Final FYP 1

  At the end of the final year project 1 we were able to propose our system diagrammatically and we had made our use case and swim-lane diagrams and selected our model and start the processing of training our model.

- Midterm FYP 2

  The plan till the mid of final year project 2 is to train our model and we have achieved it successfully.The machine learning algorithm which we are using is To train the model, K fold cross validation is employed.The data collection is essentially divided into k equal-sized subsets using this procedure.The algorithm is tested on the last fold after training on the other folds. A fresh fold is utilized as the test set after K iterations of this process, with the existing remaining (K-1) folds being used for training.

- Final FYP 2

  We will be designing an Application Programming Interface which will connect our code with our web and on the conclusion will lead us to a web based application.On the end as a final output our product will decide whether the system is being cryptojacked or not.

Chapter 6

6.1 System Flow Diagram

A system flow diagram is basically the representation of your system,that is created to show the processing and the working of the system in a simple way through the steps. All the processing measures that are taken by the system to achieve a specific goal are shown in this diagram. It shows the flow of information and the number of activities that are used to complete a specific goal.

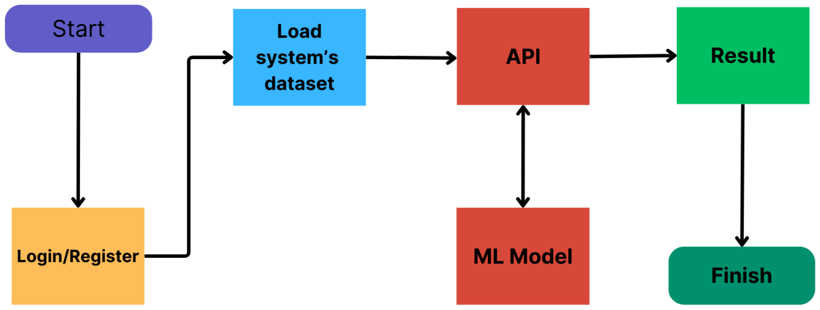


Figure 6.1: Proposed System

6.2 Timeline

Following is the timeline of Final Year Project 1 and Project 2. The timeline below shows the work division of our final year Project 1 and Project 2. All the processes are shown step by step and the working of project completion is also defined.

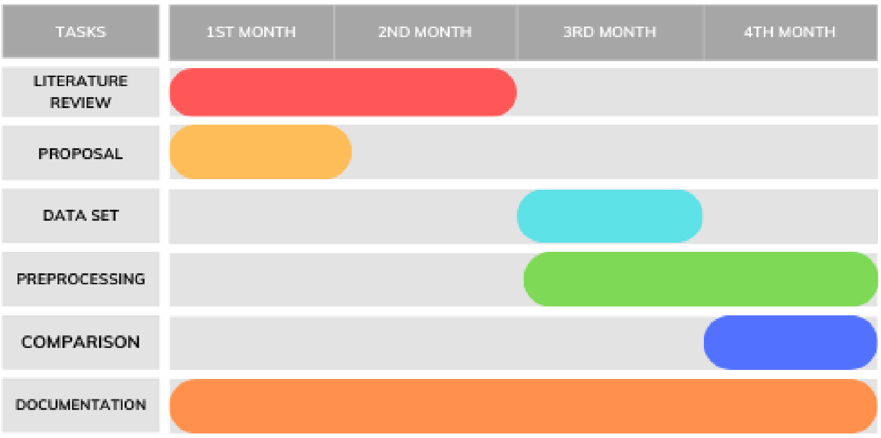


Figure 6.2: Timeline for FYP-1

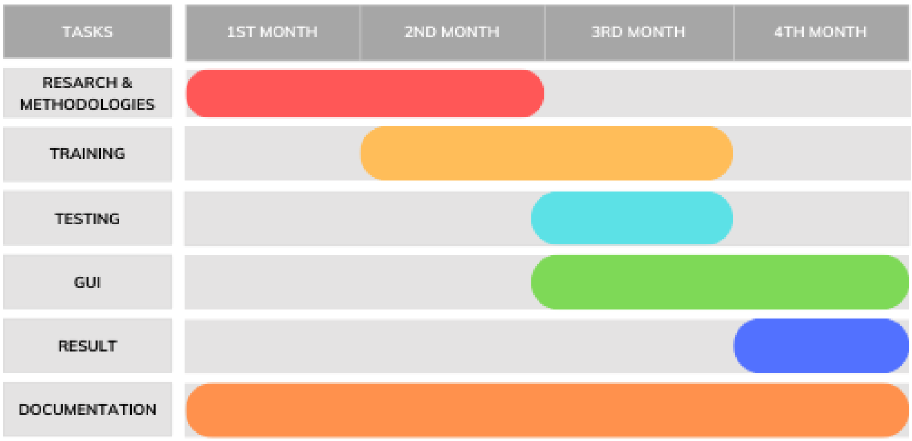


Figure 6.3: Timeline for FYP-2

Chapter 7

Iteration 2

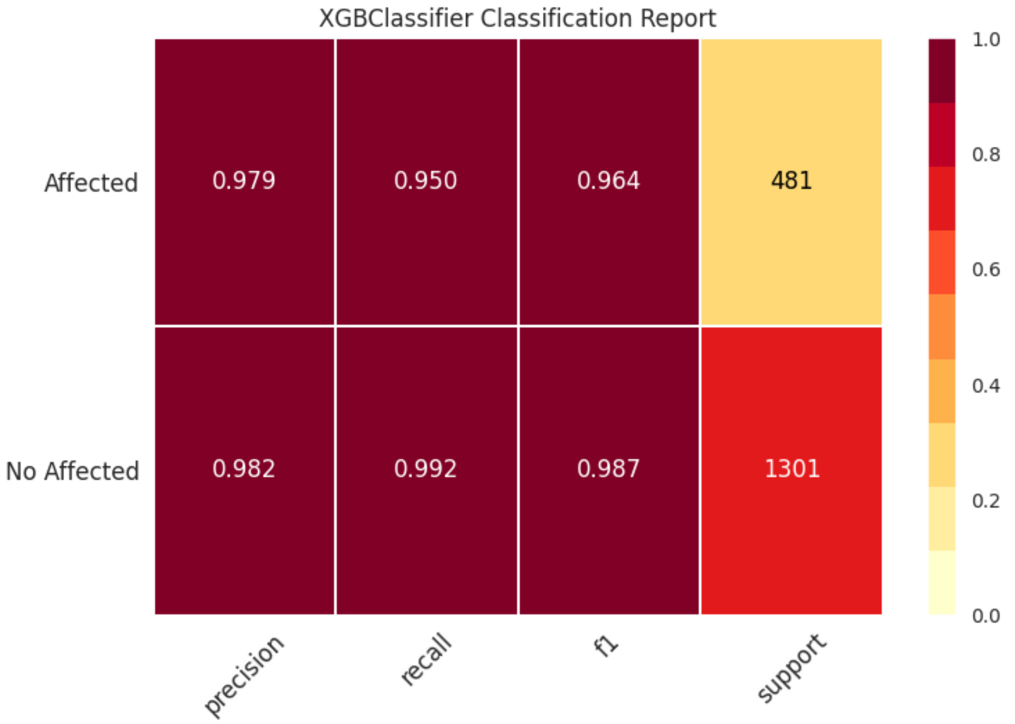
7.1 Machine Learning Model

The machine learning model used in our project is K Fold Cross Validation and we use it to avoid the manual splitting of data and to overcome the biasness in our data.It helps us with four basic steps : Data splitting ,Training and Testing,Performance Evaluation and Aggregate performance metrices.Essentially, this model automatically separates the data into k folds, one of which is selected for testing..

7.2 XGB Classifier

XGBoot classifier that is widely used for classification tasks in machine learning.XGBoot classifier is basically an implementation of the XGBoot algorithm.This classifier is used and known for its high predictive accuracy. XGBoost is a machine learning technique that leverages an ensemble of individual models, often decision trees, to generate a more reliable and precise overall prediction.XGBoost has built-in capabilities to handle missing data without the need for extensive data preprocessing.

The following diagram shows how ratio of PCs being affected and not affected correctly identified or not.



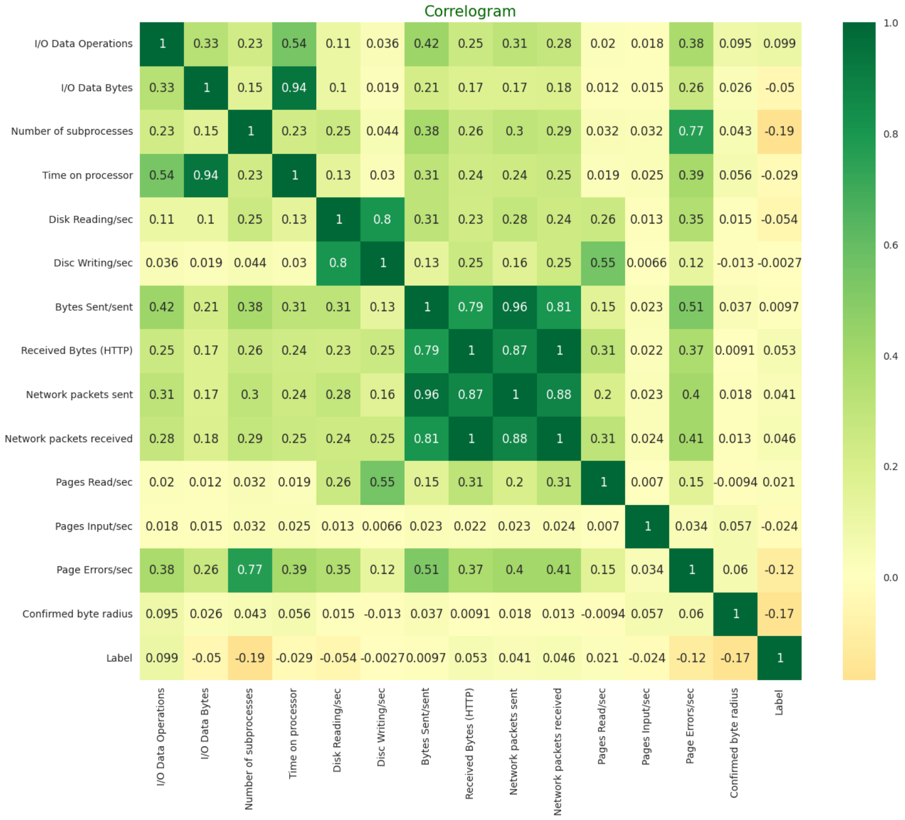
	precision	recall	f1	support
Affected	0.979	0.950	0.964	481
No Affected	0.982	0.992	0.987	1301

Figure 7.1: XGB classifier Report

7.3 System's Working

A strong API allows for the smooth integration of the main AI model with the web interface, which is being hosted on a dedicated server. The user submits the CSV files once the API has established a connection. However, before data is transmitted to the AI model, a critical preprocessing step takes place. The incoming.csv file's properties are carefully checked by the API to make sure they meet the AI model's expectations if not then accuracy of results might drop. When attribute validation is done, the API transmits the (.csv) data to the trained AI model in an easy-to-use manner. After that, the model use the classifier and its own training information to compare the values or attributes to generate a result. In its capacity as an intermediary, the API obtains the result file and dynamically displays it for user review on the webpage. Data integrity is preserved at all times during the whole procedure, protecting against irregularities and boosting the dependability of the crypto jacking detection system. User can easily engage with the web interface, knowing that the trained model's analysis will be accurate and efficient.

The diagram below shows the k-fold validation(technique) that we used in our project to make the results more accurate.



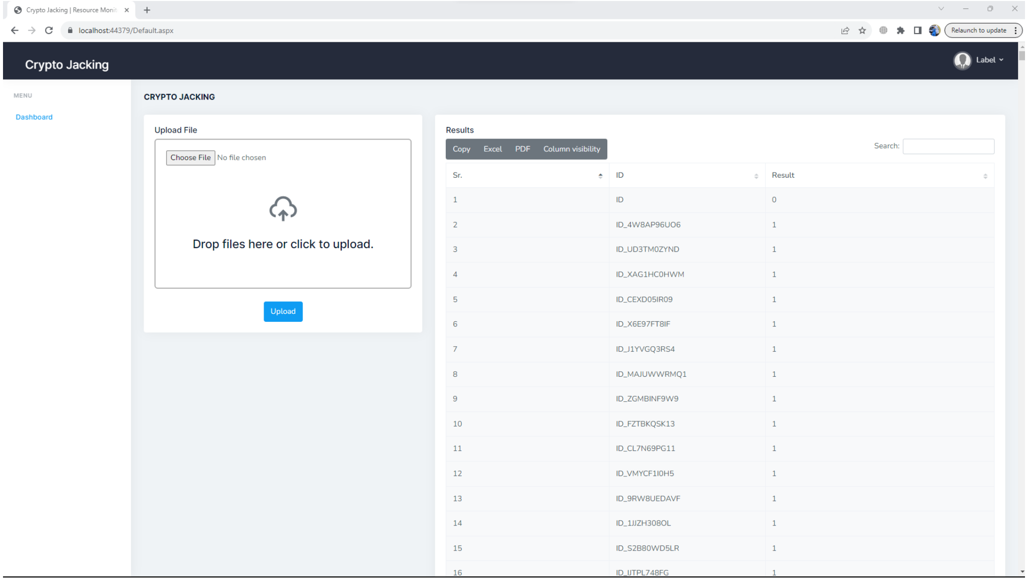
	I/O Data Operations	I/O Data Bytes	Number of subprocesses	Time on processor	Disk Reading/sec	Disc Writing/sec	Bytes Sent/sent	Received Bytes (HTTP)	Network packets sent	Network packets received	Pages Read/sec	Pages Input/sec	Page Errors/sec	Confirmed byte radius	Label
I/O Data Operations	1	0.33	0.23	0.54	0.11	0.036	0.42	0.25	0.31	0.28	0.02	0.018	0.38	0.095	0.099
I/O Data Bytes	0.33	1	0.15	0.94	0.1	0.019	0.21	0.17	0.17	0.18	0.012	0.015	0.26	0.026	-0.05
Number of subprocesses	0.23	0.15	1	0.23	0.25	0.044	0.38	0.26	0.3	0.29	0.032	0.032	0.77	0.043	-0.19
Time on processor	0.54	0.94	0.23	1	0.13	0.03	0.31	0.24	0.24	0.25	0.019	0.025	0.39	0.056	-0.029
Disk Reading/sec	0.11	0.1	0.25	0.13	1	0.8	0.31	0.23	0.28	0.24	0.26	0.013	0.35	0.015	-0.054
Disc Writing/sec	0.036	0.019	0.044	0.03	0.8	1	0.13	0.25	0.16	0.25	0.55	0.0066	0.12	-0.013	-0.0027
Bytes Sent/sent	0.42	0.21	0.38	0.31	0.31	0.13	1	0.79	0.96	0.81	0.15	0.023	0.51	0.037	0.0097
Received Bytes (HTTP)	0.25	0.17	0.26	0.24	0.23	0.25	0.79	1	0.87	1	0.31	0.022	0.37	0.0091	0.053
Network packets sent	0.31	0.17	0.3	0.24	0.28	0.16	0.96	0.87	1	0.88	0.2	0.023	0.4	0.018	0.041
Network packets received	0.28	0.18	0.29	0.25	0.24	0.25	0.81	1	0.88	1	0.31	0.024	0.41	0.013	0.046
Pages Read/sec	0.02	0.012	0.032	0.019	0.26	0.55	0.15	0.31	0.2	0.31	1	0.007	0.15	-0.0094	0.021
Pages Input/sec	0.018	0.015	0.032	0.025	0.013	0.0066	0.023	0.022	0.023	0.024	0.007	1	0.034	0.057	-0.024
Page Errors/sec	0.38	0.26	0.77	0.39	0.35	0.12	0.51	0.37	0.4	0.41	0.15	0.034	1	0.06	-0.12
Confirmed byte radius	0.095	0.026	0.043	0.056	0.015	-0.013	0.037	0.0091	0.018	0.013	-0.0094	0.057	0.06	1	-0.17
Label	0.099	-0.05	-0.19	-0.029	-0.054	-0.0027	0.0097	0.053	0.041	0.046	0.021	-0.024	-0.12	-0.17	1

Figure 7.2: Correlation Metrix

Chapter 8

User Manual

Our webpage is user friendly and hence easy to use. Simply go to the upload portion of the webpage to make use of the crypto jacking detection model. Here select the CSV file containing the attributes of the system in question so that the model can do its job. When the file is ready to be submitted, click "Upload". Click "Detect" to start the analysis process when the file has been submitted. The system will next use the train model and try to detect whether crypto jacking is being done or not. Upon completion of the analysis, the system will display the outcome as either 1 or 0. 1 denotes a possible crypto jacking activity, which simply means that additional research and action might be needed to safeguard the system. 0 indicates that no malicious activity is being done and the system resources are safe.. It is vital to remember that the quality and applicability of the data in the CSV file determines how accurate the analysis will be. For best possible outcome, users should make sure the file is converted to the necessary format. Users can consult the included documentation for any queries or problems related to the model.



Sr.	ID	Result
1	ID	0
2	ID_4W8AP96UO6	1
3	ID_UD3TM0ZYND	1
4	ID_XAG1HC0HWM	1
5	ID_CEXD05IR09	1
6	ID_X6E97FT8IF	1
7	ID_J1YVGQ3RS4	1
8	ID_MAJUWWRMQ1	1
9	ID_ZGMBINF9W9	1
10	ID_FZTBKQSK13	1
11	ID_CL7N69PG11	1
12	ID_VMYCF1I0H5	1
13	ID_9RW8UEDAVF	1
14	ID_1JJZH308OL	1
15	ID_S2B80WD5LR	1
16	ID_IJTPL748FG	1

Figure 8.1: Results on Web Page

Chapter 9

Conclusions and Future Work

As we ended the project,now we have our working web page on which once when data is uploaded.It correctly analyze the PC being cryptojacked or not.

9.0.1 Future Work

1. Model Improvement: Search for the latest machine learning algorithms or alter current ones to increase the precision and effectiveness of crypto jacking detection.

2. Investigating Features: Try varying the features or data sources in order to improve the model's capacity to distinguish between safe and harmful activity. To stay up with the latest developments in crypto jacking methods, think about dynamically modifying features.

3. Detection and Reaction in Real Time: Creating a real time detection system that keeps an eye on network activity and warn if illegal activity is detected. To fortify the defense as a whole, use automated actions like banning hostile IP addresses.

4. Machine Learning using Adversaries: Examine alternate ways to strengthen the model's defense against adversarial attacks, such as adding adversarial training to the model's

training process.

5. IP Blocking: If real time detection system works then the system can be trained to block that specific IP which was used for transferring malicious content, which ultimately resulted in system being crypto jacked.

Bibliography

[1] Caprolu. Cryptomining makes noise: Detecting cryptojacking via machine learning. *Computer Communications*, 2021.

[2] Moreno-Sancho. A data infrastructure for heterogeneous telemetry adaptation. application to netflow-based cryptojacking detection. *26th Conference on Innovation in Clouds*, 2023.

[3] Vanlioglu. The dangerous combo: Fileless malware and cryptojacking. *Mathematical Modeling and Analysis*, 2020.

[4] Xu. A novel crypto jacking covert attack method based on delayed strategy and its detection. *Digital Communications and Networks*, 2022.