

**T.C.
BAHCESEHIR UNIVERSITY
GRADUATE SCHOOL
THE DEPARTMENT OF CYBERSECURITY**

**ENHANCING THE ROBUSTNESS OF MALICIOUS URL DETECTORS
AGAINST LABEL FLIPPING ATTACKS**

**MASTER'S THESIS
NADA YOUSEF JADALLA**

ISTANBUL 2024

**T.C.
BAHCESEHIR UNIVERSITY
GRADUATE SCHOOL
THE DEPARTMENT OF CYBERSECURITY**

**ENHANCING THE ROBUSTNESS OF MALICIOUS URL DETECTORS
AGAINST LABEL FLIPPING ATTACKS**

**MASTER'S THESIS
NADA YOUSEF JADALLA**

**THESIS ADVISOR
DR. EHSAN NOWROOZI**

ISTANBUL 2024



T.C.
BAHÇEŞEHİR UNIVERSITY

GRADUATE SCHOOL

...../...../.....

MASTER THESIS APPROVAL FORM

Program Name:	CYBERSECURITY
Student's Name and Surname:	Nada Yousef Jadalla
Name of The Thesis:	ENHANCING THE ROBUSTNESS OF MALICIOUS URL DETECTORS AGAINST LABEL FLIPPING ATTACKS
Thesis Defense Date	22.01.2024

This thesis has been approved by the Graduate School which has fulfilled the necessary conditions as Master thesis.

Institute Director

Assoc. Dr. Yücel Batu SALMAN

This thesis was read by us, quality and content as a Master's thesis has been seen and accepted as sufficient.

	Title, Name	Institution	Signature
Thesis Advisor:	Ehsan NOWROOZI	Bahcesehir University	
2nd Member	Dr. Ece Gelal Soyak	Bahcesehir University	

3rd Member (Outside Institution)	Prof. Erkey Savas	Sabanci University	
---	-------------------	--------------------	--





I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: Nada Yousef Jadalla

Signature:

ABSTRACT

ENHANCING THE ROBUSTNESS OF MALICIOUS URL DETECTORS AGAINST LABEL FLIPPING ATTACKS

Nada Yousef Jadalla

Master's Program in Cyber Security

Supervisor: Dr. Ehsan NOWROOZI

January 2024, 56 pages

Malicious Uniform Resource Locators (URLs) offer adversaries the opportunity to cause potential harm to business operations across diverse sectors, encompassing transportation, healthcare, energy, and banking. Therefore, the identification of these URLs is of pivotal significance. However, existing Machine Learning (ML) models are susceptible to different cyber attacks including backdoor attacks. These attacks involve manipulating a small fraction of training data labels, such as Label Flipping (LF), where benign labels are altered to malicious ones and vice versa resulting in erroneous model behavior due to misclassification. Therefore, the integration of defense strategy into the framework of ML models becomes a necessary consideration to defend against such attacks. Through addressing the vulnerabilities of malicious URL detection by the utilization of ensemble trees specifically Random Forest (RF) in addition to highlighting the importance of deploying a defense mechanism, this study contributes to the ongoing work within security field in securing ML models against adversarial threats. We offer an innovative defense mechanism that involves an inventive alert system designed to identify the presence of flipped labels and a defense strategy crafted to unveil the true URL labels, aiming to defend against backdoor attacks on RF and XGBoost classifiers. A case study was conducted using the Alexa and Phishing Site URL datasets, revealing that LF attacks can be detected using our defense strategy. The obtained Attack Success Rate (ASR) of the applied LF attack is ranging from 50-65% within 2-5% poison rate, while the innovative defense method accurately detected flipped labels with up to 100% accuracy rate.

Keywords: AML, Backdoor Attack, Corrupted Training Sets, Cybersecurity, Poisoning Attack.

ÖZ

ENHANCING THE ROBUSTNESS OF MALICIOUS URL DETECTORS AGAINST LABEL FLIPPING ATTACKS

Nada, Jadalla

Siber Güvenlik Yüksek Lisans Programı

Tez Danışmanı: Dr. Ehsan NOWROOZI

Ocak 2024, 56 Sayfa

Kötü Amaçlı Tekdüzen Kaynak Bulucuları (URL'ler), saldırganlara ulaşım, sağlık hizmetleri, enerji ve bankacılığı kapsayan çeşitli sektörlerdeki ticari faaliyetlere potansiyel zarar verme fırsatı sunar. Bu nedenle, bu URL'lerin tanımlanması hayati öneme sahiptir. Ancak mevcut Makine Öğrenimi (ML) modelleri, arka kapı saldırıları da dahil olmak üzere farklı siber saldırılara karşı hassastır. Bu saldırılar, Etiket Çevirme (LF) gibi eğitim veri etiketlerinin küçük bir kısmının manipüle edilmesini içerir; burada iyi huylu etiketler kötü amaçlı etiketlere dönüştürülür ve bunun tersi, yanlış sınıflandırma nedeniyle hatalı model davranışına neden olur. Bu nedenle, savunma stratejisinin makine öğrenimi modelleri çerçevesine entegrasyonu, bu tür saldırılara karşı savunma için gerekli bir husus haline gelmektedir. Bu çalışma, bir savunma mekanizması konuşlandırmanın önemini vurgulamanın yanı sıra, topluluk ağaçlarının, özellikle de Rastgele Ormanın (RF) kullanılması yoluyla kötü amaçlı URL algılamanın güvenlik açıklarını ele alarak, makine öğrenimi modellerinin rakip tehditlere karşı güvenliğinin sağlanmasında güvenlik alanında devam eden çalışmalara katkıda bulunur. RF ve XGBoost sınıflandırıcılara yönelik arka kapı saldırılarına karşı savunmayı amaçlayan, ters çevrilmiş etiketlerin varlığını tespit etmek için tasarlanmış yaratıcı bir uyarı sistemi ve gerçek URL etiketlerini ortaya çıkarmak için hazırlanmış bir savunma stratejisi içeren yenilikçi bir savunma mekanizması sunuyoruz. Alexa ve Kimlik Avı Sitesi URL'si veri kümeleri kullanılarak bir vaka çalışması yürütüldü ve LF saldırılarının savunma stratejimiz kullanılarak tespit edilebildiği ortaya çıktı. Uygulanan LF saldırısının elde edilen Saldırı Başarı Oranı (ASR), %2-5 zehirlenme

oranı dahilinde %50-65 arasında deęiřirken, yenilikçi savunma yöntemi ters çevrilmiř etiketleri %100'e varan doęruluk oranıyla doęru bir řekilde tespit eder.

Anahtar Kelimeler: AML, Arka Kapı Saldırısı, Bozuk Eęitim Setleri, Siber Güvenlik, Zehirlenme Saldırısı.



ACKNOWLEDGEMENTS

I extend my deepest gratitude to Dr. Ehsan Nowroozi for his consistent guidance, counsel, constructive feedback and motivation during the entire thesis journey. Throughout the thesis process, he did not only provide the necessary support but also encouraged me to think out of the box and the completion of this thesis could not have been possible without his expertise. Dr. Nowroozi was a constant source of assistance, always available when needed, and offered unwavering support at every stage. His enduring passion for cyber security and commitment to teaching have been invaluable, contributing significantly to my academic growth. The opportunity to meet and learn from him has been a source of profound gratitude, and I appreciate his guidance that made this work possible.

I would also like to express my sincere thanks to my family, Yousef Jadalla and Omar Jadalla in addition to my beloved mother, for their exceptional support throughout my master's journey and boundless love throughout my life. Their efforts and teachings have shaped my belief in myself, and everything I have achieved today is a testament to their dedication. They have empowered me to stand independently and have played an integral role in nurturing the strength and confidence that define me as a woman.

Lastly, my sincere appreciation goes to all my fellow friends and colleagues who have been a pillar of support on this journey. Their encouragement has been invaluable, and I am grateful for the solidarity and inspiration they have provided.

TABLE OF CONTENTS

ABSTRACT.....	viii
ÖZ	ix
ACKNOWLEDGEMENTS	xi
LIST OF TABLES	xiv
LIST OF FIGURES	xv
LIST OF ABBREVIATIONS	xvii
Chapter 1	1
Introduction.....	1
1.1 Background and Motivation.....	1
1.2 Problem Statement	2
1.3 Purpose Of The Study	3
1.4 Significane of the Study	3
1.5 Organization of the Thesis	4
Chapter 2	6
Literature Review.....	6
2.1 Related Work.....	8
Chapter 3	13
Methodology	13
3.1 Data Collection.....	13
3.1.1 Dataset used.	13
3.1.2 Preprocessing steps.	14
3.2 Ensemble Tree Algorithms Selection.....	15
3.3 LF based Attack Simulation	17
3.3.1 Attacker’s goal	17
3.3.2 Attacker’s knowledge.....	17

3.3.3 Attacker's capability	17
3.4 Proposed Defense Strategy	19
3.5 Limitations	21
Chapter 4	22
Experiment	22
Chapter 5	24
Experiment Results	24
5.1 Run ML Models with Original Dataset.....	25
5.2 Apply LF attack on ML Models.....	27
5.3 Apply Defense on ML Models.....	38
5.3.1 Find Optimal K Value	38
5.3.1 Apply Defense on RF	39
Chapter 6	52
Conclusion and Future Work	52
REFERENCES.....	53

LIST OF TABLES

TABLES

Table 1 Pros and Cons of Existing Defense Methods in comparison with Our's.....	10
Table 2 Six Benign and Six Malicious URL Datasets	13
Table 3 The Utilized Six Merged Datasets	14
Table 4 RF Accuracy Rates in the Absence of RLF attack	25
Table 5 XGboost Accuracy Rates in the Absence of RLF attack.....	26
Table 6 RF accuracy rate with RLF attack.....	27
Table 7 XGBoost accuracy rate with RLF attack	28
Table 8 The Optimal Value of K for All Six Datasets	38
Table 9 RF Accuracy Rate with K-NN Defense against RLF Attack.....	39
Table 10 XGBoost Accuracy Rate with K-NN Defense against RLF Attack	50

LIST OF FIGURES

FIGURES

Figure 1 The relationship between AI, ML, NN & DL	7
Figure 2 Adversarial Attack Phases	8
Figure 3 K-NN Approach to Predict Labels.....	19
Figure 4 Example of Poisoning Attack against Classifier.	22
Figure 5 Run ML Model with Original Dataset.....	23
Figure 6 Run ML Model with Poisoned Dataset	23
Figure 11 Run RF with 2% Poisoned Dataset 1.....	30
Figure 12 Run RF with 3% Poisoned Dataset 1.....	31
Figure 13 Run RF with 4% Poisoned Dataset 1.....	31
Figure 14 Run RF with 5% Poisoned Dataset 1.....	32
Figure 15 Run RF with 2% Poisoned Dataset 2.....	32
Figure 16 Run RF with 3% Poisoned Dataset 2.....	33
Figure 17 Run RF with 4% Poisoned Dataset 2.....	33
Figure 18 Run RF with 5% Poisoned Dataset 2.....	34
Figure 19 Run RF with 2% Poisoned Dataset 3.....	34
Figure 20 Run RF with 3% Poisoned Dataset 3.....	35
Figure 21 Run RF with 4% Poisoned Dataset 3.....	35
Figure 22 Run RF with 5% Poisoned Dataset 3.....	36
Figure 23 Run RF with 2% Poisoned Dataset 4.....	36
Figure 24 Run RF with 3% Poisoned Dataset 4.....	37
Figure 25 Run RF with 4% Poisoned Dataset 4.....	37
Figure 26 Run RF with 5% Poisoned Dataset 4.....	38
Figure 27 Recovered Dataset 1 from 2% poisoned samples.....	41
Figure 28 Recovered Dataset 1 from 3% poisoned samples.....	41
Figure 29 Recovered Dataset 1 from 4% poisoned samples.....	42
Figure 30 Recovered Dataset 1 from 5% poisoned samples.....	43
Figure 31 A duplicate of Figure 27 to highlight the difference between recovered & original dataset	43
Figure 32 Recovered Dataset 2 from 2% poisoned samples.....	44
Figure 33 Recovered Dataset 2 from 3% poisoned samples.....	44

Figure 34 Recovered Dataset 2 from 4% poisoned samples.....	45
Figure 35 Recovered Dataset 2 from 5% poisoned samples.....	45
Figure 36 Recovered Dataset 3 from 2% poisoned samples.....	46
Figure 37 Recovered Dataset 3 from 3% poisoned samples.....	46
Figure 38 Recovered Dataset 3 from 4% poisoned samples.....	47
Figure 39 Recovered Dataset 3 from 5% poisoned samples.....	47
Figure 40 Recovered Dataset 4 from 2% poisoned samples.....	48
Figure 41 Recovered Dataset 4 from 3% poisoned samples.....	49
Figure 42 Recovered Dataset 4 from 4% poisoned samples.....	49
Figure 43 Recovered Dataset 4 from 5% poisoned samples.....	50



LIST OF ABBREVIATIONS

ML	Machine Learning
LF	Label Flipping
RLF	Random Label Flipping
URLs	Uniform Resource Locators
DL	Deep Learning
K-LID	K – Local Intrinsic Dimensionality
PEFL	Privacy-enhanced Federated learning
DT	Decision Tree
MLP	Multi-layer Perceptron
HAR	Human Activity Recognition
IoT	Internet of Things
C&C	Command & Control
ASR	Attack Success Rate
CNN	Convolutional Neural Network
RSRNN	Regularized Synthetic Reduced Nearest Neighbor
LSD	Label-based Semi-supervised Defense
CSD	Clustering-based Semi supervised Defense
KNN	K-Nearest Neighbor
RF	Random Forest
XGBoost	Extreme Gradient Boosting
TPR	True Positive Rate
TNR	True Negative Rate
FPR	False Positive Rate
FNR	False Negative Rate

Chapter 1

Introduction

1.1 Background and Motivation

In the landscape of internet and the World Wide Web, URLs are important for the seamless functioning over the internet and play an essential role to browse online. However, these URLs have also become a potential target for attackers to impact businesses across diverse sectors in addition to active users. To address such threat, several research has been conducted in the application of ML security systems, including RF models (Nowroozi E. a., 2022). Yet, ML models remain vulnerable to backdoor attacks due to their susceptibility to poisoned training data. RF, a widely adopted ensemble learning technique used in several fields, is capable to work with datasets that includes feature interactions, complex and abnormal data. Nevertheless, RF is susceptible to adversarial attacks, such as poisoning attacks. There are different approaches that can be applied by the attacker to poison URLs and fool detection systems such as RF with the aim to avoid being detected and appear as a non-malicious URL. One of the approaches is through applying a label flipping attack on the URLs labels by flipping malicious labels denoted as “1” to benign and benign labels denoted as “0” to malicious. A high threat is caused by leading the user to a malicious web to access systems without authorization and cause system failures. There are various papers and software solutions, including Symantec Endpoint Protection (Symantec, 2023), McAfee (Mcafee, 2023), Cisco Umbrella (Cisco, 2023), which focuses on detecting malicious URLs. Also, malicious URLs are detected through the usage of common blacklists which involves most updated benign and malicious URLs, (Sankaran, 2021). However, the focus of current solutions does not involve the integration of malicious URL defense strategy within its framework. Proposing a defense strategy is essential and should be integrated into all ML models. Blacklisting falls short in identifying unknown or new malicious URLs, and attackers adeptly change URL characteristics, adapting their attack strategies to evade detection and effectively injecting poisoned data into training datasets.

1.2 Problem Statement

Technology has evolved massively over the recent years and now stands as an indispensable necessity in today's world, particularly in the era of the Internet of Things (IoT). The internet serves as the connective fabric for myriad devices, enhancing our lives and simplifying everyday tasks. However, this technological interconnectedness also exposes all systems and devices, including Artificial Intelligence (AI) and ML systems, to potential security threats. Attacks on ML systems are categorized into Poisoning (Causative) Attacks or Evasion (Exploratory) Attacks and the type of attack can be determined depending on which phase that attack is applied (Ahmed, 2021), (Nowroozi E. a., 2023). For instance, poisoning attacks occur during the training phase, aiming to manipulate, or fail the ML system such as (Lin, 2021):

- Clean Label Attacks
- LF attacks
- Backdoor attacks

On the other hand, evasion attacks are executed during the testing phase seeking to produce malicious data without altering with the ML model, including (Moisejevs, 2019):

- Gradient Attacks,
- Confidence Score Attacks
- Hard Label Attacks

With the escalating number of attacks on ML systems, diverse research studies have explored strategies to implement poisoning attacks, including LF attacks under different threat models. Typically, worst-case scenarios are examined, assuming the attacker can directly poison the training dataset to impact the model's behavior. For example, (Chang, 2020) orchestrated a data poisoning attack on a RF-based ML model to diminish its classification accuracy, whereas (Dunn, 2020) introduced poisoning attacks using label modification functions on both ML and Deep Learning (DL) models, employing varying poison rates to measure performance degradation. Over the years, researchers in the ML field have primarily concentrated on addressing all existing detection solutions (Carlini, 2017) and work on improving them (Gilmer,

2018), as cyber-attacks evolve with increasingly sophisticated strategies. A comprehensive survey conducted by (Kumar, 2020) aimed to identify and address gaps in protecting ML systems across different organizations, revealing a prevalent threat from poisoning attacks. Despite the significance of security to the ML operations, organizations often neglect the use of appropriate security tools. This lack of awareness is evident as ML system usage proliferates in the software industry (Kumar, 2020). Major organizations like Google and Microsoft have advocated for initiatives to secure ML systems (Google, 2023), (Advertisement, 2023) and numerous studies have focused on applying poisoning attacks and constructing defense strategies through various methods e.g. K-LID (Shahid, 2022), PEFL (Liu, 2021).

1.3 Purpose Of The Study

This study is conducted with the purpose to investigate, emphasize, and enhance the robustness of RF and XGBoost classifiers when LF attack is executed using manipulated training datasets in the field of ensemble tree-based URL classification. Our objective is to demonstrate the efficacy of the proposed defense strategy. An innovative defense strategy is proposed through an exhaustive analysis of LF attack strategies and their repercussions on ensemble tree models. This method, a pioneering advancement, serves as the inaugural mechanism capable of distinguishing malicious URLs from benign ones by detecting manipulated labels, unveiling their actual label, and ultimately bolstering the resilience of RF-based malicious URL detection systems. By filling the missing gaps of backdoor attacks and ensemble tree classifiers, our study makes a substantial contribution to the domain of ML security. It advances our understanding of strong countermeasures to safeguard against sophisticated threats encountered in real-world cybersecurity scenarios. Our Python simulation code is publicly accessible on Github (E.Nowroozi, 2023). In summary, the study stands as a cornerstone in advancing our understanding of ML security threats, addressing RF and XGBoost vulnerabilities against LF attacks, providing defense solution, bridging critical gaps, and positively impacting ML security field.

1.4 Significance of the Study

The significance of this study is rooted in its exceptional contribution to the dynamic field of ML security, specifically within the domain of ensemble tree-based URL classification. By investigating, highlighting, and mitigating the vulnerabilities

introduced by LF attacks on RF and XGBoost classifiers using corrupted training sets, this study significantly advances our knowledge on ML security challenges and vulnerabilities. This study serves as an essential step forward by addressing the vulnerabilities of ensemble tree models when exposed to LF attacks specifically in today's evolving nature of cybersecurity threats demands further exploration and studies in this ML security field.

A main aspect of the study's significance is the development of an innovative defense mechanism specifically developed to counter the unique challenges exposed by LF attacks on RF and XGBoost classifiers. By proposing a mechanism capable of not only identifying flipped labels but also identifying their true labels, the research introduces a groundbreaking approach that goes beyond theoretical contributions. The ultimate goal of enhancing the robustness of RF-based malicious URL detection systems sets this study apart by offering a practical solution to a critical problem in the field of ML security. This study not only contributes to theoretical knowledge but also provides a concrete defense strategy that aligns with the practical needs of the ML security field.

Furthermore, the study's contribution extends to bridging the critical gap between backdoor attacks and ensemble tree classifiers. By exploring the intersection of these elements, the research provides valuable insights into the security challenges faced by ensemble tree classifiers when exposed to poisoning attacks specifically LF attacks. This bridge is essential for developing effective countermeasures against sophisticated threats, ensuring that the study's findings hold broader implications for the development and deployment of secure ML models. This study contributes not only to academic knowledge but also to the ongoing efforts to secure digital environments against evolving cyber threats.

1.5 Organization of the Thesis

This study consists of six chapters. Chapter 1 presents the introduction, which covers the main contributions, motivations, and rationale of our research. Our primary objective is to propose a detection and defense method against adversarial attacks, in specific, LF attacks that target RF and XGBoost classifiers. The remaining chapters are organized as follows: Chapter 2 presents the related work within our domain based

on the conducted literature review. Chapter 3 explains the applied methodology in our study including the used datasets, selected classifier, applied LF-based attack, and proposed defense method against such attacks. Chapter 4 and Chapter 5 describes how the experiment is implemented and presents the experimental results of running clean datasets, poisoned datasets, and recovered datasets. Finally, Chapter 6 concludes our research and discusses future work.



Chapter 2

Literature Review

A detailed literature review is conducted before this thesis is written. Although considerable studies have been conducted on the security of ensemble trees and the ML security field, there needs to have more researches on mitigating LF attacks in malicious URL detectors using ensemble trees. This literature review is focused explicitly on below research topics;

- Overview of ML, Ensemble Trees and RF.
- Robustness of Ensemble based ML
- Applications of RF based ML models and their vulnerabilities.
- Types of Data Poisoning attacks and their application
- LF Attacks against malicious URL detectors.
- Defending against data poisoning attacks and strategies studied before.
- Data sanitization against LF attacks
- Computing Attack success rate of LF attacks
- Addressing different defense strategies to enhance the RF model behavior.

The most common words during the literature review are ML, RF, LF, Malicious URL Detectors, Training set, ASR, training accuracy, poisoning attack, K-NN. Therefore, some need to understand those concepts and at least have the basic knowledge about them to comprehend the impact of LF attacks and propose defense strategy to mitigate the impact in Malicious URL Detectors Using Ensemble Trees.

Many may think that ML is a new topic that has been introduced in the recent years however this goes back to the 17th century where scientists were discussing the automation of the human thinking process and the ideas kept on growing slowly until the 1950s when Alan Turing introduced the theoretical aspects of AI as explained by (Kuipers, 2022). The concept of AI expanded throughout the years into the field of ML, Neural Networks (NN) and DL as shown in Figure 1.

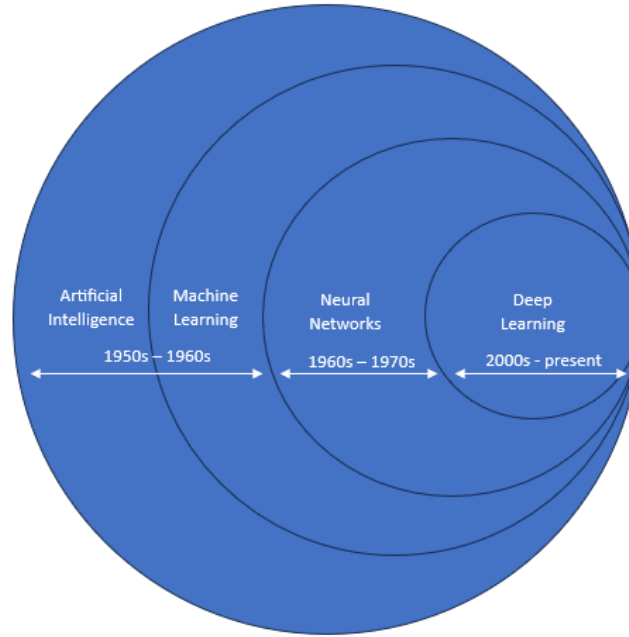


Figure 1 The relationship between AI, ML, NN & DL

Over the years, there has been a discernible evolution in the tactics employed by attackers to compromise ML systems. As ML technology becomes more prevalent in various applications, including cybersecurity, attackers are mainly focused on exploiting vulnerabilities in these systems to achieve their malicious objectives. A significant way to implement an attack on ML systems is through the manipulation of training data, where attackers inject subtly crafted instances to deceive ML models during the learning process. These attacks, commonly known as adversarial attacks, can compromise the integrity and effectiveness of ML models across different domains.

In security-critical field, such as ML security, adversarial attacks pose a severe threat to cybersecurity (He, 2023). In the context of image recognition, for example, attackers may introduce imperceptible perturbations to images, leading the model to misclassify them. Similarly, in malicious URL detectors, attackers can manipulate the training dataset by flipping the labels of both benign and malicious URLs with the aim to impact the behavior of the classifier. As shown in Figure 2, the adversarial attack phases can target the training dataset or the testing dataset depending on the applied type of attack. A LF attack is a type of poisoning attacks that targets the training dataset during training phase.

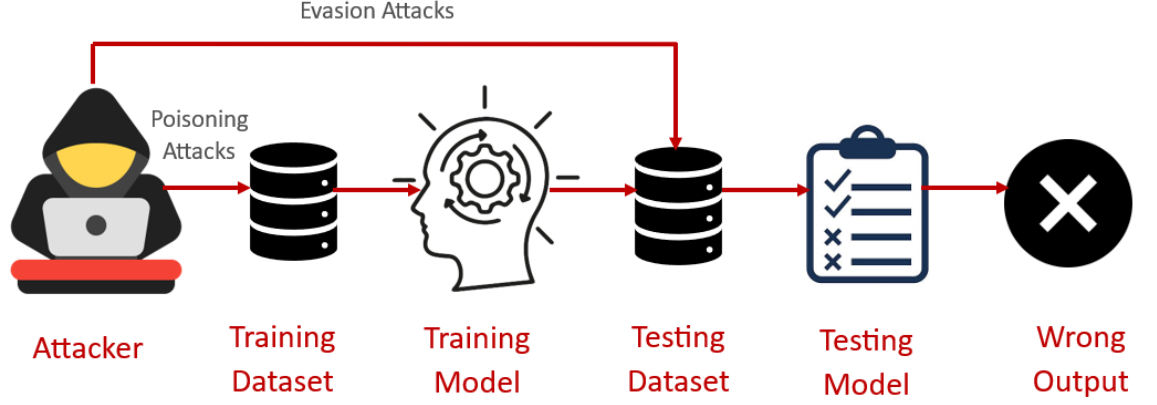


Figure 2 Adversarial Attack Phases

Furthermore, attacks can be classified into targeted attacks, where the attacker changes the behavior of the model on particular instances, or into untargeted attacks, where the attacker aims to impact a model's performance with random instances or scenarios (Shahid, 2022). Accordingly, a LF attack can be a targeted attack on selected data or can manipulate the training dataset randomly. Both ways aim to impact the training accuracy and fool the classifier. These attacks challenge the robustness of ML systems in specific RF or XGBoost and underscore the need for advanced defenses.

2.1 Related Work

With the surge in digitization and widespread technology adoption, ML models find themselves vulnerable to poisoning attacks including LF attacks, emerging as a significant research area in recent years (Goldblum, 2022). For instance, (Shahid, 2022) built a white-box attack on a Human Activity Recognition (HAR) system that execute the attack by randomly altering labels from the training dataset using different ML models like RF, Multi-layer Perceptron (MLP), XGBoost, and Decision Tree (DT). This attack minimizes the loss function on true data, but noticeable changes in the data limit the attacker's capacity to cause extensive damage to the model. Whereas (Ramirez, 2022) executed untargeted and targeted LF attacks on various ML models, including DT and RF models. Both attacks degraded the accuracy of all ML models and increased misclassification rates, requiring access to the dataset during the training phase to execute the attack. Another study explored poisoning attacks against IoT fake packet classifiers, using label modification functions on RF-based ML models with different poison rates to illustrate performance degradation (Dunn, 2020). (Yerlikaya,

2022) applied a Random LF (RLF) attack on different ML models, including the RF model, aiming to impact model accuracy. RF exhibited a minor accuracy drop within a 0-12.5% poisoning rate, but a further increase in poison rate resulted in higher accuracy degradation, indicating the model detected the poisoned labels. The current research focus emphasizes the development of model-agnostic defenses to identify poisoned data and defend against poisoning attacks targeting diverse ML models. (Gardiner, 2016) proposed several Command & Control (C&C) detection systems to identify sophisticated attacks and reduce false positive rates. However, storing all network traffic becomes challenging as network size increases, though it remains a crucial requirement for most C&C detection methods. (Paudice A. a.-G., 2019) aims to detect poisoned labels using the K-NN approach and mitigate LF attack impacts through label sanitization. However, this approach assumes the availability of a large number of benign samples for sanitization, potentially reducing classifier accuracy. Another proposed scheme employs outlier detection to identify attack points targeting linear classifiers (Paudice A. a.-G., 2018). Yet, applying this scheme in large, high-dimensional datasets poses challenges. Conversely, (Chan, 2021) explores how detecting causative attacks can enhance learning robustness through a two-step secure classification model using data complexity measures. These measures calculate classification difficulty by assessing the geometrical characteristics of data. Moreover, (Shahid, 2022) discusses the utilization of a K-NN defense scheme to predict true labels and recover most data, resulting in a significant boost in model accuracy. (Anisetti, 2022) proposes a hash-based ensemble approach to enhance the robustness of RF models, emphasizing the importance of ensemble size to prevent accuracy drops due to oversized ensembles. (Taheri, 2020) presents a Label-based Semi-supervised Defense (LSD) and a Clustering-based Semi-supervised Defense (CSD) that increase model accuracy by identifying poisoned samples and using clustering techniques to recover true labels. However, these methods exhibit relatively weaker performance compared to other defense mechanisms.

(Cheng, 2021) introduces a boosting ensemble mechanism for two classifications, involving the training of several weak classifiers on the same dataset to combine into a single robust classifier. However, the model is highly sensitive to noisy and abnormal data which considered as a vulnerability, making it susceptible to the impact of such data. Meanwhile, (Tavallali, 2022) employs the Regularized

Synthetic Reduced Nearest Neighbor (RSRNN) defense mechanism, examining samples to identify any exceeding the confidence range, categorizing them as malicious. RSRNN successfully achieved the smallest test error while detecting a significant portion of malicious samples. Yet, a challenge reminiscent of the *K*-means problem persists in learning an SRNN model—the identification of the most relevant *K* value of samples that yields the lowest error rate. Table 1 provides a summary of diverse detection and defense methods designed to safeguard ML models against LF attacks.

Table 1

Pros and Cons of Existing Defense Methods in comparison with Our's

Reference	Model	Utilized Dataset	Pros	Cons
(Gardiner, 2016)	Several classifiers	10 real world datasets	-Ability to identify sophisticated attacks -Decreases the false positive rate	- High traffic storage requirement
(Paudice A. a.-G., 2019)	ML: Linear Classifier	MNIST (Deng, 2012), Spambase (Hopkins, 1999) and BreastCancer (Wolberg W. M., 1995)	-Efficient against LF attacks -Can be used for different use cases.	- Impact the accuracy of the classifier
(Paudice A. a.-G., 2018)	ML: Linear Classifier	MNIST (Deng, 2012), Spambase (Hopkins, 1999) and BreastCancer (Wolberg W. M., 1995)	-Identifies flipped labels and eliminates outliers	- Outlier estimation requirement.

		M., 1995)			
(Chan, 2021)	Ensemble Trees	UCI ML Repository (UCI, 2023) & KEEL-dataset Repository (Alcal-Fdez, 2011)	-Enhance model robustness -Applicable for several types of attacks	-Cannot identify the flipped labels -Trusted data requirement for training	
(Shahid, 2022)	Ensemble Trees	HAR dataset (Reyes-Ortiz, 2012)	-Identify the true label of poisoned data -Enhance accuracy rate	-Effeciency limitation	
(Anisetti, 2022)	Ensemble Trees: RF	Musk2 (Chapman, 1994) and Android malware	- Obtained high accuracy in the presence of various perturbations.	- The size of the ensemble must be taken into account and adjusted based on the adversary's level of knowledge	
(Cheng, 2021)	Ensemble Trees: Adaboost	Spambase (Hopkins, 1999), Breast-w (Wolberg W. , 1992), Kr-vs-kp (Shapiro, 1989) and so forth	-Merges number of weak classifiers to create a strong classifier	- Sensitivity against abnormal & noisy data	
(Tavallali, 2022)	Ensemble Trees: Adaboost	MNIST (Deng, 2012)	-Applicable to a range of ML models -Ability to identify	-Learning challenge -Limitation on the LF budget	

				manipulated labels		
(Taheri, 2020)	Neural Network: CNN	Drebin (2014), Contagio (Contagio, 2023), and Genome (Zhou, 2012)	(Arp, -Targets malware identification -Limited applicability to alternative platforms, and the defense method may not extend to unfamiliar malware samples.	IoT	- An innovative clustering-based attack method is proposed	
This Thesis	Ensemble Trees: RF	URL Datasets in Table 3	-Very detection rate of flipped labels -URLs Dataset are considered for the first time in computer networks	high	- Trusted data requirement for identifying K value. - Challenge when implementing with feature poisoning attack.	

Chapter 3

Methodology

In order to accomplish all-inclusive research, brief information about the utilized datasets, LF application, and defense strategy used in the experiment in addition to the limitation of our study will be summarized in this section. Before the experiment section, the reader will be introduced to the used techniques and general information about the subject will be given.

3.1 Data Collection

3.1.1 Dataset used.

A set of 3,980,870 benign and malicious URLs is used in this study (Nowroozi E. a., 2022), collected from group of datasets listed in Table 2. This group of twelve datasets includes six benign URL datasets and six malicious URL datasets. Afterwards, we formed six datasets from merging the benign and malicious datasets in Table 2.

Table 2

Six Benign and Six Malicious URL Datasets

No.	Benign URL Datasets	Malicious URL Datasets
1	Pristine Alexa	Phishing Site URL
2	Pristine Crowdfunder	Phishtank
3	Pristine DMOZ	Malicious data URL
4	Benign set URL	ISCX-URL-2016
5	Non-Malicious URL	Phishstorm
6	Pristine ISCX	Malicious set URL

Six merged datasets are displayed in Table 3 includes benign URLs labeled as '0' and malicious URLs labeled as '1'. A comprehensive statistical analysis was conducted to further understand these datasets. The study revealed that malicious URLs within the selected six datasets exhibit a broader range of variability in their length compared to URLs classified as benign, as identified in (Nowroozi E. a., 2022). For our specific poisoning scenario this is not considered as a challenge as

the focus is on manipulating the labels of both benign and malicious URLs, introducing a nuanced dimension to the analysis.

Table 3

The Utilized Six Merged Datasets

Dataset Number	Merged Datasets
Dataset 1	Pristine Alexa and Phishing Site URL
Dataset 2	Pristine Crowdfunder and malicious Phishtank
Dataset 3	Pristine DMOZ and Malicious data URL
Dataset 4	Benign set URL and Malicious ISCX
Dataset 5	Non-Malicious URL and Malicious Phishstorm
Dataset 6	Pristine ISCX and Malicious set URL

3.1.2 Preprocessing steps.

To attain precise results, it is imperative to adjust and restructure the dataset to conform to the required input format for our dependencies. The preprocessing phase commenced with the comprehensive gathering of data from six benign and six malicious datasets, ensuring the inclusion of a representative set of both malicious and benign URL datasets. Following this, the interquartile range technique was employed to rescale the data. This process involved the removal of all duplicates and unprocessed cells containing values, along with the exclusion of repetitive hostname URLs. Subsequently, the collections of URLs underwent a shuffling process. Samples were then extracted from these datasets to facilitate a more in-depth analysis. Consequently, the datasets are now prepared for utilization with ML models such as RF.

The importance of preprocessing extended to feature engineering, a key step in boosting the discriminative power of the RF and XGboost model. Relevant features, such as domain length, character frequencies, and structural components, are extracted from the URLs. The dataset is then normalized and scaled to mitigate the influence of varying scales on the behaviour of the RF and XGBoost classifiers, optimizing their ability to handle diverse and complex features. Furthermore, the dataset passes through a stratified splitting process, creating distinct training and testing sets while preserving the distribution of classes in both subsets. In this study, we split the six datasets into

79% for training set and 21% for testing set.

These preprocessing steps, designed with a deep understanding of both the RF and XGBoost classifiers, aimed to refine the dataset, optimize feature representation, and establish a robust foundation specifically tailored for the intricacies of RF-based malicious URL detection systems.

3.2 Ensemble Tree Algorithms Selection

In this study we have chosen two types of classifiers: RF and XGBoost model. The RF is a type of supervised learning that relies on multiple decision trees to validate a single output. The underlying principle of RF is established in the recognition that while individual trees may make accurate predictions, there is a risk that some trees could excessively adapt themselves to specific data samples, resulting in overfitting. To address this concern, the approach involves merging the outcomes of various trees, each excelling and overfitting in many ways, successfully minimize the impact of overfitting. The overfitting existence can be eliminated without impacting the predictive capability of the trees. The RF model includes several parameters that can be modified such as n estimators, max features, and prepruning settings like max depth. A larger value for n estimators is recommended to enhance the RF performance. Increasing the number of trees being averaged makes the ensemble more robust, effectively reducing the overfitting. However, it is crucial to acknowledge that employing more trees also entails greater memory and training time requirements. Mathematically we can represent RF with the following parameters:

- D : refers to the original dataset.
- T : indicates the number of decision trees in the RF.
 - Every tree t randomly draws a subset D_t from the training data D , usually through bootstrapping. Bootstrapping is a statistical resampling method by which subsets of data are constructed by randomly sampling with replacement from the original dataset.
- F_t : represents a random subset of features selected for each node of the tree, which is a subset of the total features F available in the dataset.
- t : denotes the decision tree constructed using the dataset D_t and features F_t with reference to a given measure.

Every tree t in the prediction phase in the ensemble independently set a class for the variable or identifies a target variable. Within the context of classification tasks, the class, y_c , is established by a majority vote among the trees indicateing the mode of class predictions and can be mathmetically illustrated as the following:

$$y_c = Mode(y_t) \text{ for } t = 1, 2, \dots, T \quad (1)$$

While for regression tasks, the identified value, y_r , is found by estimating the average value of every tree and can be illustrated in mathematical form as the following:

$$y_r = \frac{1}{T} \sum_{t=1}^T y_t \quad (2)$$

The aggregation of predictions from multiple randomized decision trees helps mitigate overfitting and enhances the performance of the RF model. RF is a powerful and widely used ML algorithm due to its robustness and effectiveness, making it applicable in various domains, including classification, regression, and feature importance analysis. The selection of RF is very suitable since the utilized six datasets are a combination of numerical and lexical features, decision trees are selected for their ability to handle multiclass classification.

The XGBoost, which stands for eXtreme Gradient Boosting, is a powerful and efficient classifier used for both regression and classification tasks. It is an implementation of the gradient boosted decision trees introduced for speed and performance. XGBoost has proven its ability to handle large datasets which is suitable for our study and gained popularity for its high predictive accuracy. XGBoost can be represented mathematically by the following objective function for a regression task:

$$Objective = \sum_{i=1}^n L(y_i, \hat{y}_i) + \sum_{k=1}^K \Omega(f_k) \quad (3)$$

For a classification task, the objective function extends to incorporate a nested summation over both training samples and classes, using binary indicators $y_{i,k}$ and predicted probabilities $y'_{i,k}$ and can be represented mathematically as the following:

$$Objective = \sum_{i=1}^n \sum_{k=1}^K L(y_{i,k}, y'_{i,k}) + \sum_{k=1}^K \Omega(f_k) \quad (4)$$

3.3 LF based Attack Simulation

As a form of poisoning attack, LF attack is applied in this study, where the attacker poisons the training dataset by flipping the benign or malicious labels of specific instances in class one to another. This attack can be executed using either a targeted LF attack or an RLF attack (Jebreel, 2022). In this study, an RLF attack is applied to the six datasets with the goal of manipulating them by randomly altering both benign and malicious labels. RLF attacks fall under the category of adversarial attacks, where the attacker aims to exploit vulnerabilities in the RF and XGBoost model by manipulating the input data during the training phase. Adversarial attacks encompass the following three main types:

- Black box.
- Gray box.
- White box attacks.

In a black box attack, the attacker lacks access to the system, representing a worst-case scenario in terms of system accessibility. In a white box attack, the attacker has full access to the classifier, including its weights and parameters. In this study, a gray box attack is implemented, where the attacker has limited access to the classifier during the training phase. The threat model of such attacks is associated with the attacker's goal, knowledge, and capability to impact the training dataset (Wang, 2022).

3.3.1 Attacker's goal. Specifies the security violation and the desired error the attacker aims to apply. In this study, the goal is to manipulate the training dataset used by the RF and XGBoost model and deceive the model by concealing the poisoned labels.

3.3.2 Attacker's knowledge. The degree of knowledge is established by the attacker's access to the feature space, the target classifier, the model parameters, and the training dataset. In this study, the attacker only has access to the training dataset; hence, the attacker's knowledge is categorized as limited knowledge.

3.3.3 Attacker's capability. This refers to the extent of control that the adversary has over both the training and testing data, as defined in (Nowroozi E. a., 2022).

By randomly altering both benign and malicious labels in five scenarios with poisoning rates of 2%, 3%, 4%, and 5%, the RLF attack is implemented in this study. Our attack method is detailed in Algorithm 1 in the appendix A. As inputs, this algorithm takes the classifier (RF or XGBoost), original dataset D , and a specified number of poisoned samples based on the poisoning rate. With reference to the given poison rate, X samples are randomly chosen from original dataset D to flip their labels and cause manipulation. The original training dataset is represented as the following:

$$D = \{(x_i, y_i)\}_{i=1}^N; \quad (5)$$

where;

x_i refers to the input data samples;

y_i refers to the true labels;

N represents the overall number of URLs.

In a RLF attack, an attacker randomly flips a subset of the true labels as y'_i which is the poisoned label saving it into a poisoned dataset that is denoted mathematically as the following:

$$D' = \{(x_i, y'_i)\}_{i=1}^M; \quad (6)$$

where;

x_i refers to the input data samples;

y'_i refers to the poisoned labels;

M indicates the number of URLs with flipped label.

The flipped labels causes a manipulation into the dataset that involves changing randomly y_i to y'_i , causing the model to learn the potentially erroneous labels.

In this study, LF attack is executed by flipping benign and malicious labels randomly on each X selected row in the original dataset D . Labels denoted as "0" are flipped to "1" and treated as malicious samples in the training, and labels denoted as "1" are flipped to "0" and treated as benign samples. After applying the RLF attack, the flipped labels are stored into the poisoned dataset D' . The RF and XGboost are trained using the poisoned dataset D' and then the accuracy rate of testing and training

is recorded. The attack is triggered by testing the saved model through flipping the whole labels within the testing dataset, measuring the accuracy rate of the testing dataset. The reported accuracy rate in this scenario reflects the ASR of labels that are flipped from the adversary's intended RLF attack approach.

3.4 Proposed Defense Strategy

The detection and defense method introduced in this study aims is designed to detect flipped labels and provide a recovered dataset sanitized from LF attacks. This method operates by taking K value, original datasets, D , and untrusted datasets, D' , as input. It uses the K -NN method shown in Figure 3 like the approach applied in (Paudice A. a.-G., 2019) to identify flipped labels by LF attack. This method involves training a model using the provided inputs to predict the true label of a given URL sample and subsequently checks if it matches the current label of the URL sample.

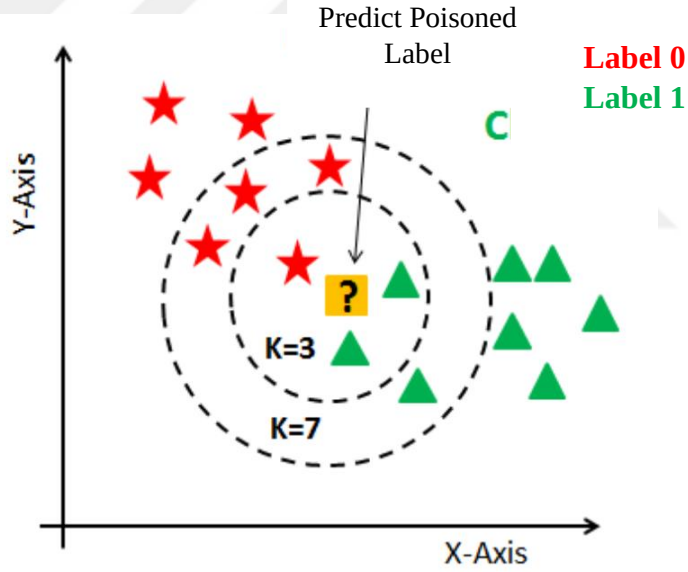


Figure 3 K -NN Approach to Predict Labels

The building of our defense strategy is constructed considering the following two steps:

3.4.1 Choose the optimal K value. For optimal performance, the selection of the K value is crucial to prevent overfitting or underfitting of the model. The chosen K value dictates the number of labels to be examined and aids in predicting the value of the tested label. Algorithm 2 in appendix B, provided in the appendix, determines

the best K value within the X set of K values. It selects K based on the minimum error rate. This algorithm takes as input the following:

- D : original dataset,
- D' : untrusted (poisoned) dataset,
- X : set of K values.
 - We considered a specific X set of K values for evaluation including the following range of values: $X = \{1, 3, 5, 7, \dots, 35, 37, \text{ and } 39\}$

For every K value in the above X set of values, the algorithm calculates the distance d of labels in D from the tested label in D' . It then identifies the mode of the K nearest labels to validate the label value. The selection of the most relevant K value occurs when the number of mismatch labels, Lm , in D is reduced to zero, thereby minimizing the error rate. This step is crucial for ensuring the accuracy and reliability of the model in detecting and addressing LF attacks.

3.4.2 Defense method. In this study, we considered K -NN as our defense method to mitigate the impact of RLF attack. The K -NN algorithm can be expressed as follows: Given a dataset D with data points represented as vectors in a d -dimensional feature space, let x_i and y_i be the target data points for which we want to make a prediction or classification and n is the total number of labels in the dataset. The K -NN algorithm calculates distances between x_i and all other data points in D , typically utilizing metrics like the Euclidean or Manhattan distance, defined mathematically as follows:

$$d = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}; \quad (7)$$

Our detection and defense method is outlined in Algorithm 3 in the appendix C. This algorithm takes as an input the following parameters:

- D : original dataset,
- D' : untrusted dataset,
- K : number of nearest neighbors.

Following the selection of the value of K using Algorithm 2 in the appendix, this method computes the distance d of each label y_i in the original dataset with

reference to the input sample x_i . It then selects the K nearest samples and the algorithm determines the mode of K nearest labels. It checks if mode equals 1, then the predicted label, x_p is equal to 1; otherwise if mode equals 0 then, x_p equals 0. Subsequently, x_p is compared with the value of x_i and if they are not equal to each other, the system detects the LF attack, activates an alarm to notify the user and rectifies the true label by saving x_p into x_i . The predicted label value is determined based on the mode of the K nearest labels, which is identified by sorting the numbers from lowest to highest and selecting the most frequently occurring label. For instance, if the mode of K nearest labels equals to “1”, then the predicted label will equal to “1”, and if the mode equals to “0”, then the predicted label will equal to “0”. Lastly, the method checks if the predicted label matches the value of the label in the untrusted dataset D' . If a mismatch is detected, the label is corrected, and an alarm is activated to alert the user about the LF attack. This process repeats until all labels in the dataset are checked, and if any poisoned label is identified, the label is corrected, and an alarm is triggered to inform the user about the LF attack. Algorithm 3 in the appendix delineates the steps of the implemented defense mechanism.

3.5 Limitations

While the proposed K -NN defense strategy provides promising results in mitigating the impact of LF attacks on malicious URL detectors, it is essential to acknowledge and highlight the limitations within the design of K -NN defense model. One significant constraint is the dependency on the availability of a trusted dataset during the construction of the K -NN model.

The effectiveness of the K -NN approach relies deeply when identifying the best value of K on the availability and quality of the trusted dataset used for RF and XGboost models training. However, to build a robust defense mechanism, the challenge here arises when access to a such comprehensive and trusted dataset is limited or when the dataset itself is subject to any type of manipulation.

Chapter 4

Experiment

In order to assess the transferability of LF attacks against RF and XGBoost classifiers, we have twelve datasets, these include six benign and six malicious datasets. A RLF scenario is performed as explained in Section 3.2 to fool the RF and XGBoost classifiers and impact their behaviour.

In RF and XGBoost models, the attacker can impact the system performance by manipulating the training dataset and this is considered as an attack influence. RLF attack is a type of poisoning attack that takes place during the training phase and is categorized under causative attacks. Figure 4 demonstrates how the attack affects the training phase by flipping the benign labels into malicious labels and the malicious labels into benign, forcing the model to misclassify the output.

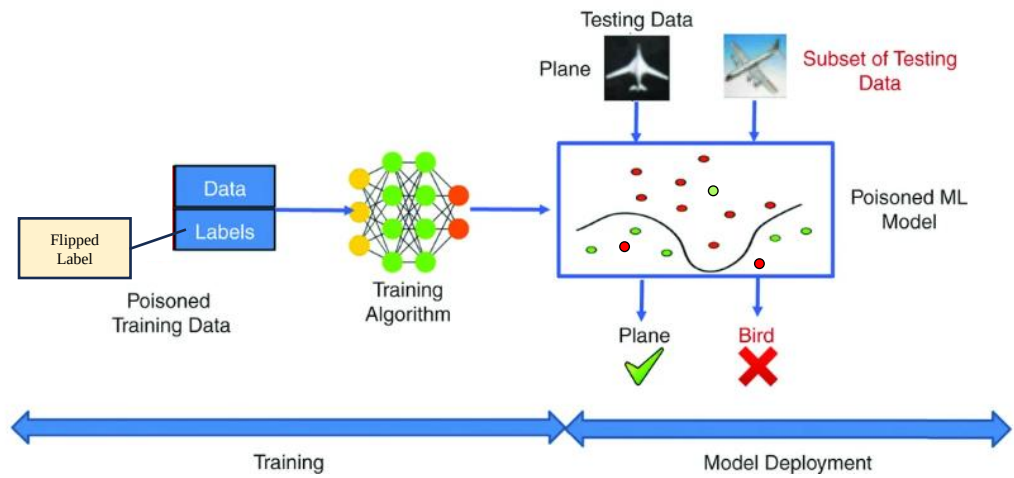


Figure 4 Example of Poisoning Attack against Classifier.

Before applying the RLF attack similar to Figure 4, we ran both RF and XGBoost classifiers with the original six datasets and recorded the accuracy as illustrated in Figure 5 to evaluate the model's performance when no attack is applied and set a baseline to assess the attack impact and efficiency of the proposed defense mechanism.

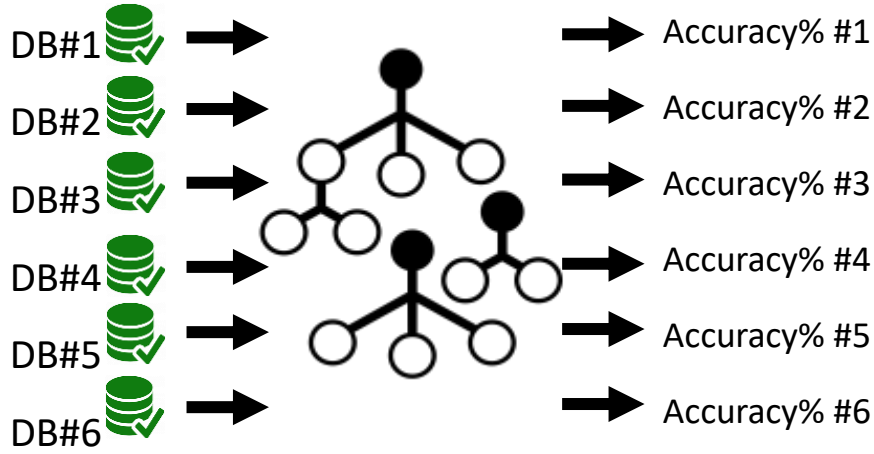


Figure 5 Run ML Model with Original Dataset

Then we applied RLF on six combined datasets by flipping both the labels of benign URLs and the labels of malicious URLs with the aim to impact the performance of the RF and XGboost classifiers as illustrated in Figure 6. In addition we computed the ASR for each attack scenario through testing the stored poisoned model by flipping the entire labels within the testing dataset to assess the success of the applied attack.

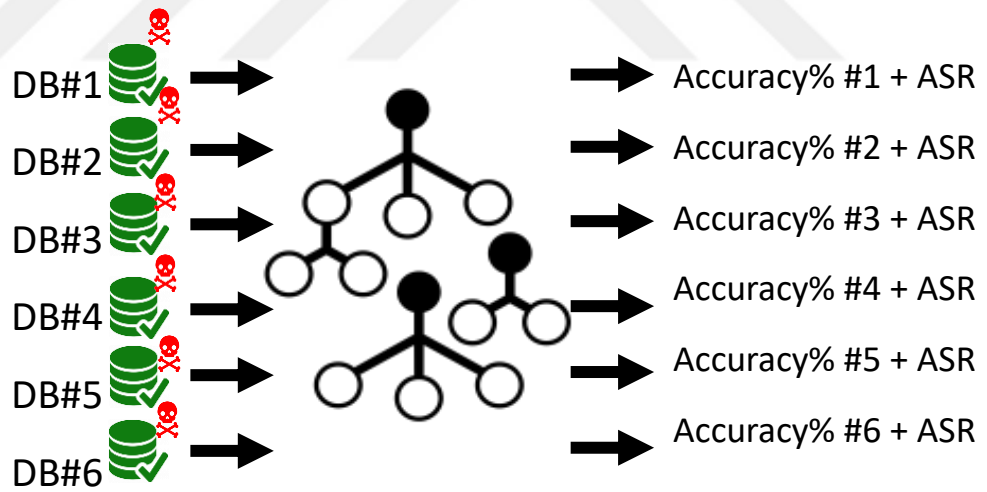


Figure 6 Run ML Model with Poisoned Dataset

After successfully fooling the classifier, we proposed a defense strategy based on the *K-NN* approach to mitigate the impact of RLF attack and provide the true label of each poisoned label then we calculated the testing accuracy rate of each scenario to evaluate the enhancement of the RF and XGBoost model performance. This finalizes the experiment process of our study.

Chapter 5

Experiment Results

In our experimental setup, we used the six different datasets outlined in Table 3 (Nowroozi, 2022). 1000 malicious and benign URLs are listed in each of the six datasets. The malicious URLs labels are denoted as “1” while benign URLs labels are denoted as “0”. We initiated our experiment without any attack by running the original dataset D on RF and XGBoost classifiers. Afterwards, a RLF attack is executed on each dataset by flipping benign labels denoted as “0” to “1” and malicious labels denoted as “1” to “0”. The attack is applied using different poison rates to assess the performance of RF and XGBoost against such attacks. To facilitate the assessment of the RF and XGBoost performance, we employed a confusion matrix that consist of set of metrics including:

- A True Positive Rate (TPR): represents the proportion of actual positive samples correctly identified by the ML model as true. In our case, it is the positive URL correctly classified by RF and XGBoost as the true class. TPR depends on the TP samples and FN samples and is mathmetically calculated as follows:

$$TPR = \frac{TP}{TP+FN} \quad (8)$$

- A True Negative Rate (TNR): represents the proportion of actual negative samples correctly identified by the ML model as false. In our case, it is the negative URL samples that is classified by RF and XGBoost as the true class. TNR depends on the TN samples and FP samples and is mathmetically calculated as follows:

$$TNR = \frac{TN}{TN+FP} \quad (9)$$

- A False Positive Rate (FPR): represents the proportion of negative samples misclassified by the ML model as positive. In our case: it includes all negative URL samples that RF and XGBoost misclassified as positive samples. FPR depends on the FP samples and TN samples and is mathmetically calculated as follows:

$$FPR = \frac{FP}{FP+TN} \quad (10)$$

- A False Negative Rate (FNR): represents the proportion of positive samples misclassified by the ML model as negative. In our case, it includes all positive URL samples that RF and XGBoost misclassified as negative samples. FNR depends on the FN samples and TP samples and is mathematically calculated as follows:

$$FNR = \frac{FN}{FN+TP} \quad (11)$$

Utilizing the confusion matrix, various performance metrics were computed using specific formulas to gain insights into the effectiveness of the RF and XGBoost classifiers. Accuracy is determined by the total number of examples that can be predicted with high reliability, relative to all examples in the six dataset. The calculation of accuracy rate relied on the TP, TN, FP and FN as shown below:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (12)$$

5.1 Run ML Models with Original Dataset

In addition to determining accuracy for individual datasets, we evaluate the effectiveness of both the adversary and the ML model including RF and XGBoost through the calculation of ASR. ASR quantifies the model's accuracy rate when subjected to testing on a poisoned testing dataset. This section provides a comprehensive analysis of the experimental results for each scenario, offering insights into the model's performance, the influence of the applied attack, and the efficacy of our proposed defense method against RLF attacks.

To assess the efficiency of LF attack and the proposed defense mechanism, a baseline mode is established by training both the RF and XGboost model on six original datasets outlined in Table 3 to evaluate RF accuracy rate without applying any type of attack. The accuracy rates derived from training and testing the RF model with original datasets are presented in Table 4.

Table 4

RF Accuracy Rates in the Absence of RLF attack

Utilized Dataset	Training Acc.	Testing Acc.
<i>Dataset#1</i>	100.00%	100.00%
<i>Dataset#2</i>	100.00%	100.00%
<i>Dataset#3</i>	100.00%	100.00%
<i>Dataset#4</i>	100.00%	100.00%
<i>Dataset#5</i>	100.00%	100.00%
<i>Dataset#6</i>	99.87%	100.00%

And the accuracy rates derived from training and testing the XGBoost model with original datasets are presented in Table 5.

Table 5
XGboost Accuracy Rates in the Absence of RLF attack

Utilized Dataset	Training Acc.	Testing Acc.
<i>Dataset#1</i>	100.00%	99.72%
<i>Dataset#2</i>	100.00%	99.72%
<i>Dataset#3</i>	100.00%	99.72%
<i>Dataset#4</i>	100.00%	99.72%
<i>Dataset#5</i>	100.00%	99.72%
<i>Dataset#6</i>	100.00%	99.72%

The results distinctly show that in the absence of an attack on the RF model, there is no notable accuracy reduction for five datasets in the RF scenario and for the six datasets in the XGBoost scenario. Only Dataset 6 exhibits a negligible decrease with RF model, resulting in an accuracy rate of 99.87%. Nonetheless, this decrement is considered insignificant within the realm of ML systems. It is crucial to acknowledge that the acceptable threshold for accuracy reduction can vary, contingent on the specific application, domain, and potential repercussions of errors.

Taking the six original datasets on the RF model into consideration serves as a dependable benchmark for assessing both attack and defense scenarios in the next subsections, allowing the evaluation of model performance. The objective of our study is to introduce a defense mechanism against RLF attacks capable of detecting manipulated URL labels and bolstering the robustness of the RF model.

5.2 Apply LF attack on ML Models

As shown in Algorithm 1 in the appendix, we initially introduce a poison rate into each dataset ranging from 2% to 5%, resulting in 16 to 40 URL labels being poisoned. The outcomes of our RLF attack on RF are documented in Table 6.

Table 6

RF accuracy rate with RLF attack

Utilized Dataset	Poison Rate	Flipped Labels	Training Acc.	ASR
<i>Dataset#1</i>	2%	16	98.35%	57.52%
	3%	24	97.09%	50.47%
	4%	32	95.95%	61.09%
	5%	40	97.34%	55.23%
<i>Dataset#2</i>	2%	16	97.97%	58.10%
	3%	24	96.96%	55.23%
	4%	32	96.08%	50.47%
	5%	40	95.18%	64.76%
<i>Dataset#3</i>	2%	16	98.23%	50.48%
	3%	24	97.22%	53.33%
	4%	32	96.20%	56.19%
	5%	40	95.19%	55.23%
<i>Dataset#4</i>	2%	16	97.97%	55.24%
	3%	24	97.47%	54.28%
	4%	32	96.33%	58.09%
	5%	40	95.19%	56.19%

<i>Dataset#5</i>	2%	16	97.97%	62.38%
	3%	24	96.96%	53.33%
	4%	32	96.08%	57.14%
	5%	40	95.19%	61.09%
<i>Dataset#6</i>	2%	16	98.10%	59.05%
	3%	24	96.96%	58.09%
	4%	32	96.08%	52.38%
	5%	40	95.23%	56.19%

The results indicate the effectiveness of the employed attack, as it had a minimal 2% impact on the RF model when 2% of the data was poisoned. Even with a slight escalation in the poison rate (from 2% to 5%), the RF model's accuracy drop was limited to no more than 5%, and the impact was not deemed significant enough to critically influence the model's accuracy. Similarly, we applied the RLF attack on XGBoost and the obtained results are outlined in Table 7.

Table 7

XGBoost accuracy rate with RLF attack

Utilized Dataset	Poison Rate	Flipped Labels	Training Acc.	ASR
<i>Dataset#1</i>	2%	16	98.09%	54.32%
	3%	24	97.30%	61.89%
	4%	32	96.19%	52.43%
	5%	40	95.08%	55.95%
<i>Dataset#2</i>	2%	16	98.25%	58.11%
	3%	24	97.14%	57.30%
	4%	32	96.67%	56.49%
	5%	40	95.07%	55.41%
<i>Dataset#3</i>	2%	16	98.41%	56.49%
	3%	24	96.98%	57.57%
	4%	32	96.19%	56.49%
	5%	40	95.39%	56.76%
<i>Dataset#4</i>	2%	16	98.09%	58.65%
	3%	24	96.98%	58.65%

	4%	32	96.66%	54.32%
	5%	40	95.23%	54.32%
	2%	16	97.93%	54.86%
<i>Dataset#5</i>	3%	24	97.14%	52.97%
	4%	32	96.19%	54.59%
	5%	40	94.92%	59.19%
	2%	16	97.93%	55.95%
<i>Dataset#6</i>	3%	24	96.98%	55.41%
	4%	32	96.34%	55.14%
	5%	40	95.23%	57.03%

This approach maintained the covert nature of the attack, allowing poisoned backdoor samples to be seamlessly integrated into the training dataset. We determine the success of the applied attack by measuring the ASR through flipping all labels within the testing datasets to identify the model's classification error rate. While normal testing accuracy measures correct classification, ASR focuses on the classification error. The ASR values exceeded 40%, signifying the method's success in evading detection by the URL detector and acting as a ghost attack, remaining unnoticed within the system, aligning with the predetermined threshold of the study. For instance, as shown in Table 6, Dataset 1 achieved a training accuracy of 98.35% and an ASR of 57.62% with a 2% poisoning rate, while Dataset 2 achieved a training accuracy of 97.97% and an ASR of 58.1% with a 2% poisoning rate. Both results highlight the attacker's success in manipulating Dataset 1 and Dataset 2, effectively deceiving the RF model with only 2% poisoned data.

For in-depth analysis, visual plots are generated for the poisoned Datasets 1, 2, 3, and 4 to visually represent the flipped labels. A red threshold, functioning as a decision margin, is introduced to distinguish between URL labels denoted as "0" and URL labels denoted as "1". Figures 11-26 showcase the visual representation of Dataset 1, 2, 3, & 4 after randomly flipping 2-5% of the labels, affecting both benign and malicious URLs. For example, in figure 14, the impact of 5% poisoned data is evident as 40 manipulated samples are inaccurately positioned with respect to the decision margin.

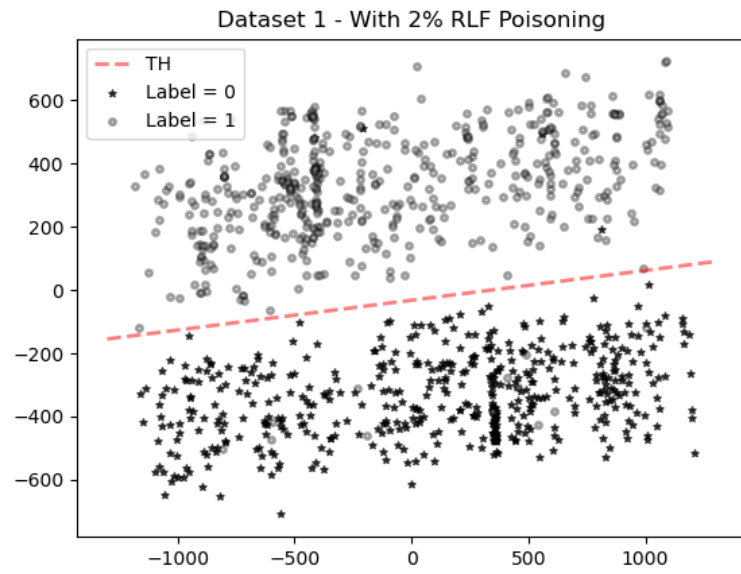


Figure 7 Run RF with 2% Poisoned Dataset 1

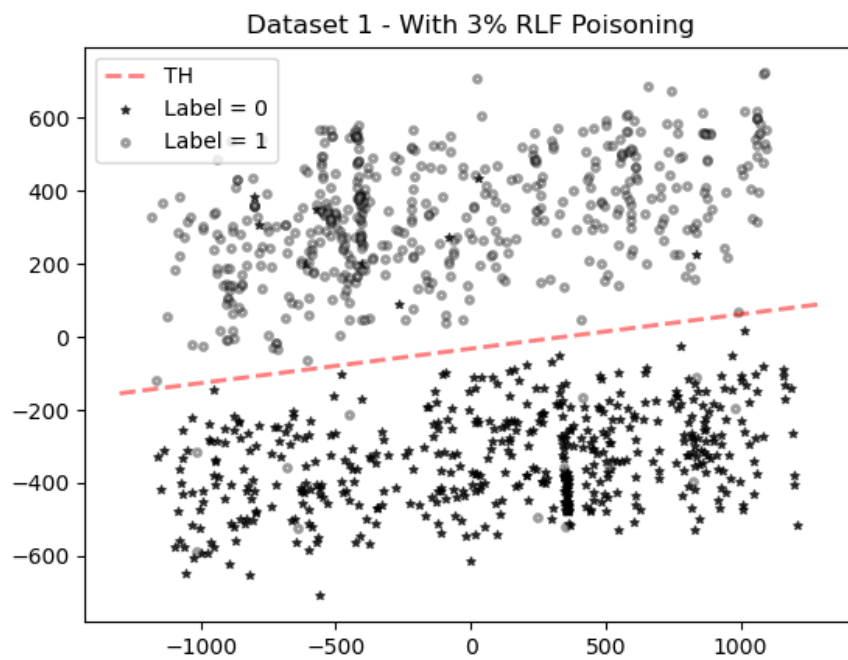


Figure 8 Run RF with 3% Poisoned Dataset 1

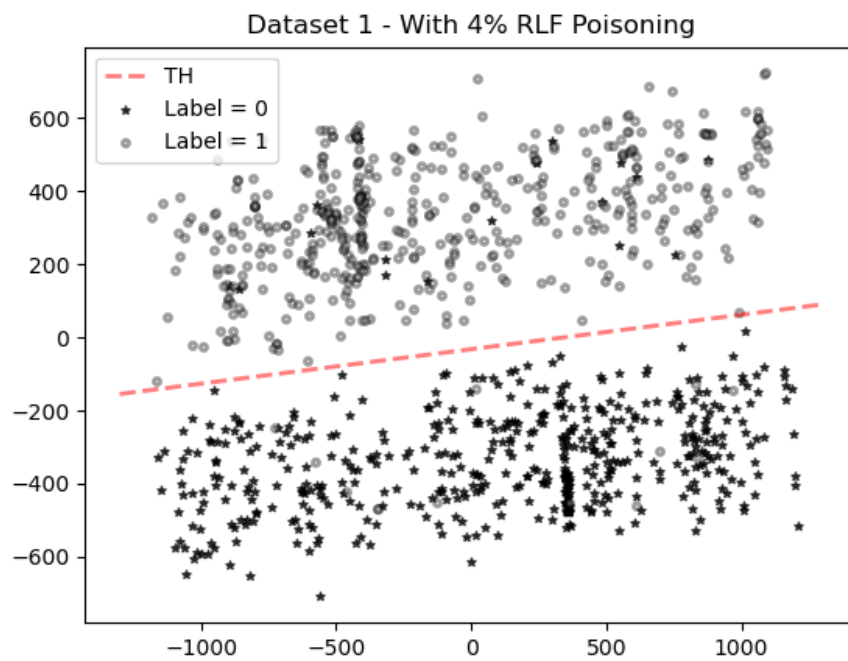


Figure 9 Run RF with 4% Poisoned Dataset 1

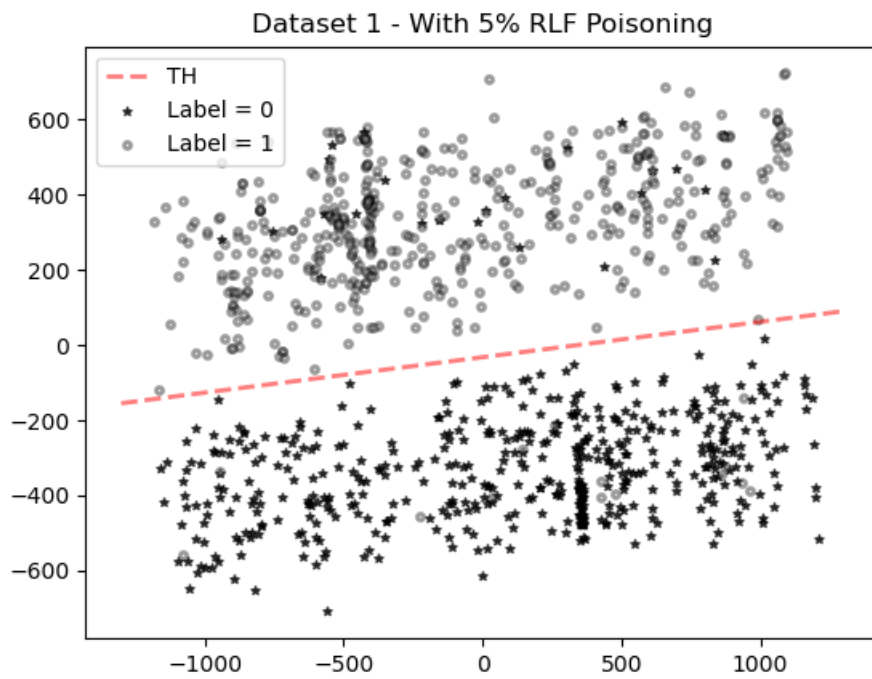


Figure 10 Run RF with 5% Poisoned Dataset 1

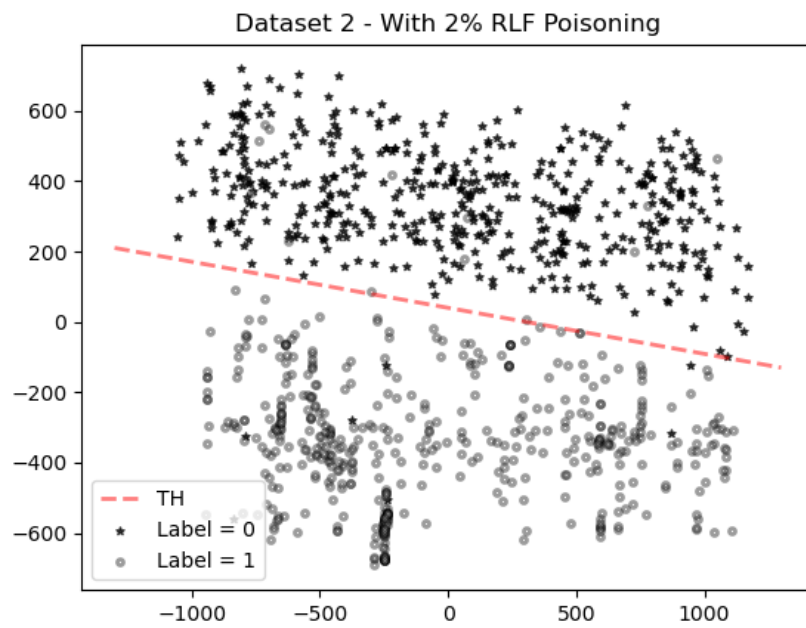


Figure 11 Run RF with 2% Poisoned Dataset 2

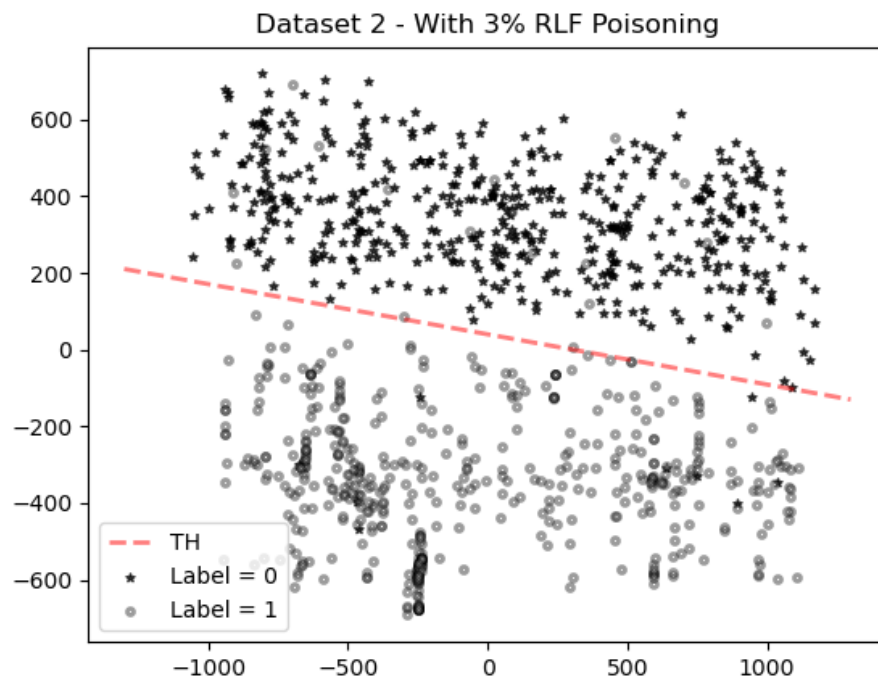


Figure 12 Run RF with 3% Poisoned Dataset 2

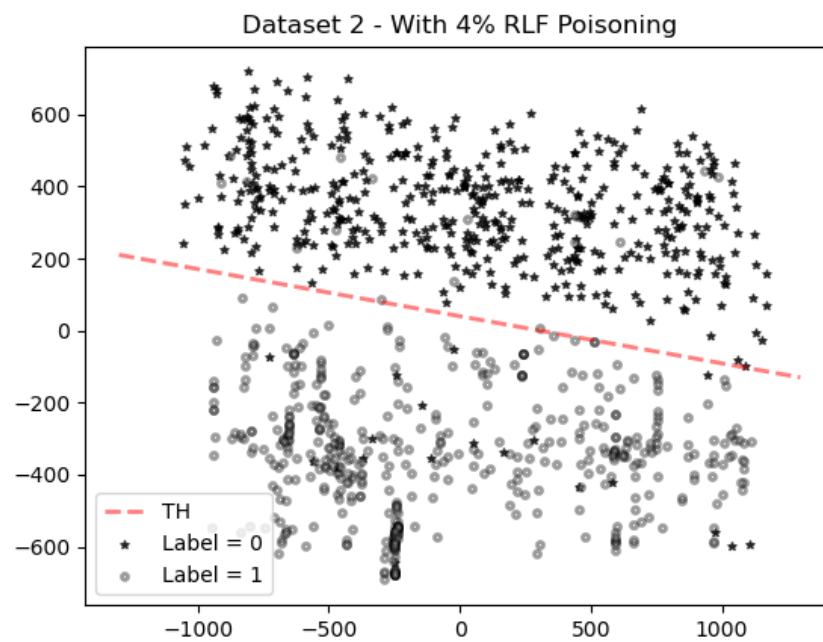


Figure 13 Run RF with 4% Poisoned Dataset 2

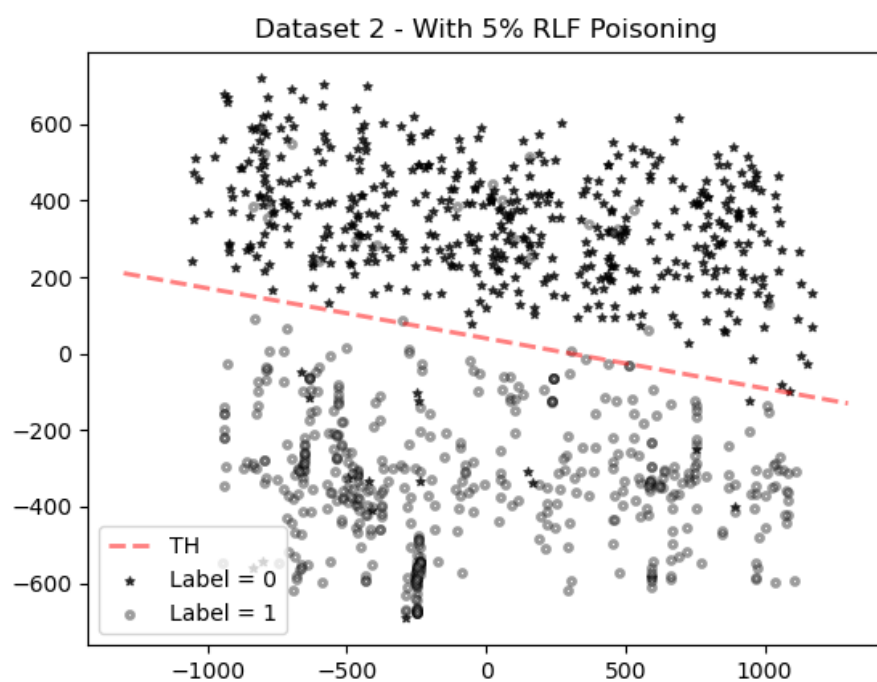


Figure 14 Run RF with 5% Poisoned Dataset 2

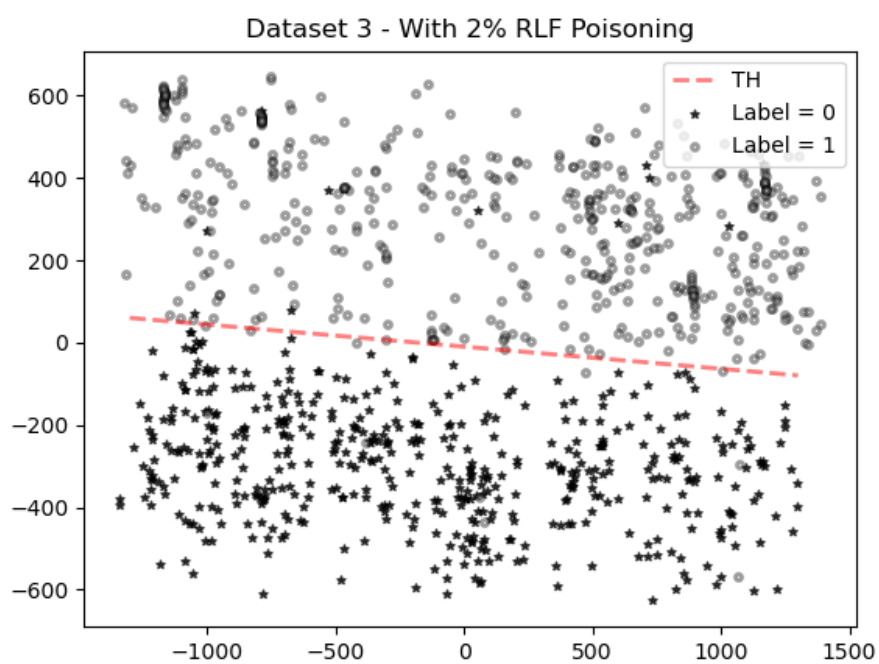


Figure 15 Run RF with 2% Poisoned Dataset 3

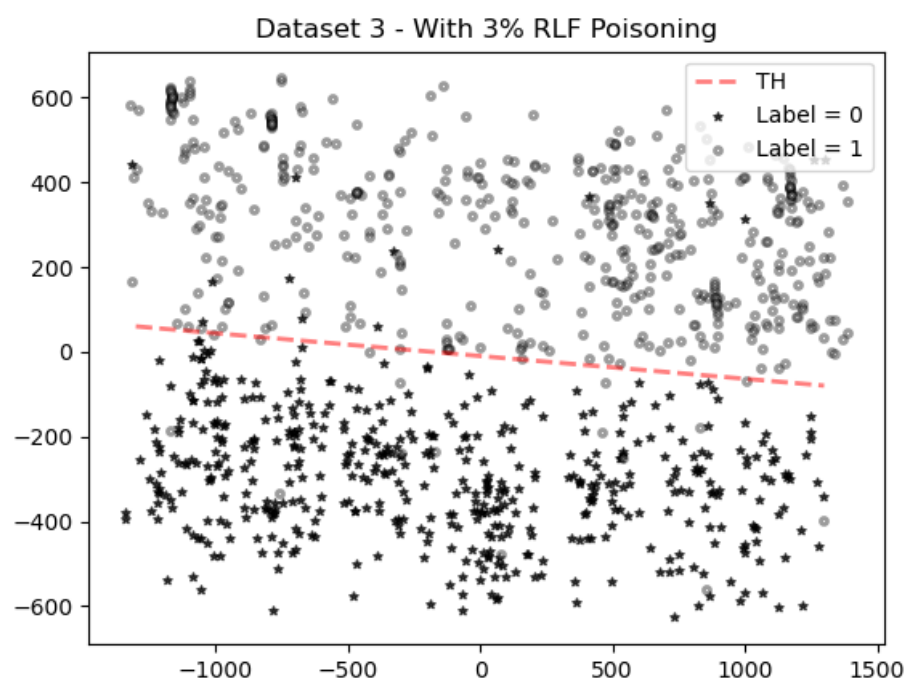


Figure 16 Run RF with 3% Poisoned Dataset 3

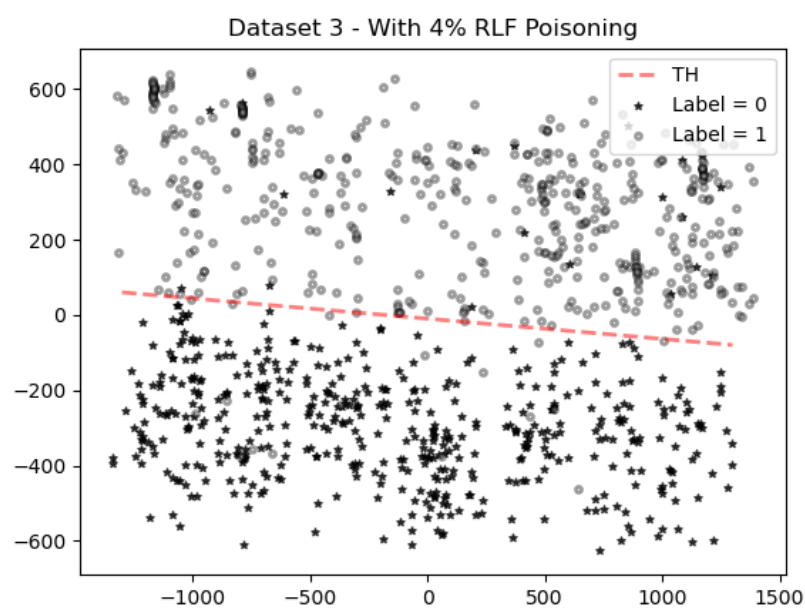


Figure 17 Run RF with 4% Poisoned Dataset 3

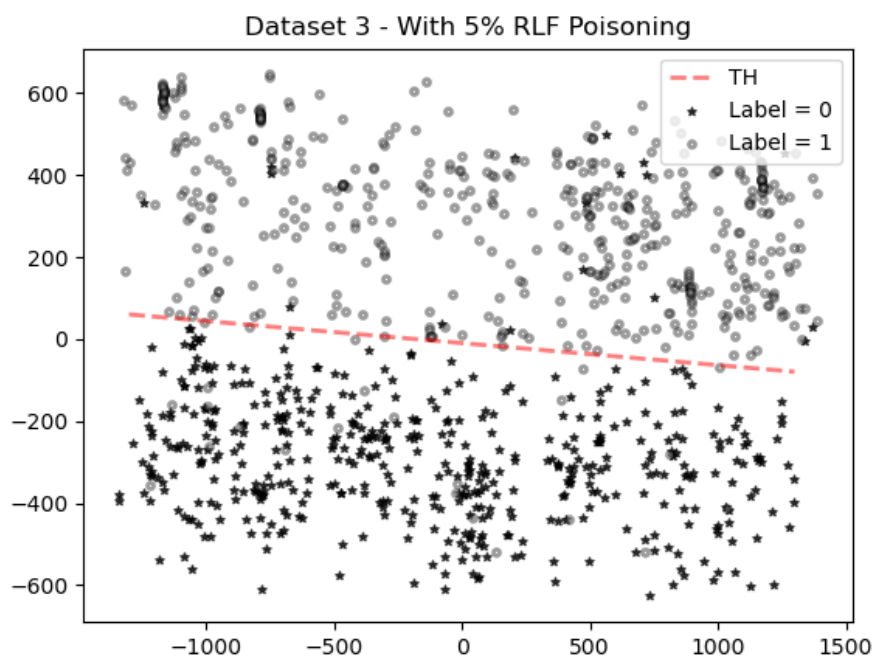


Figure 18 Run RF with 5% Poisoned Dataset 3

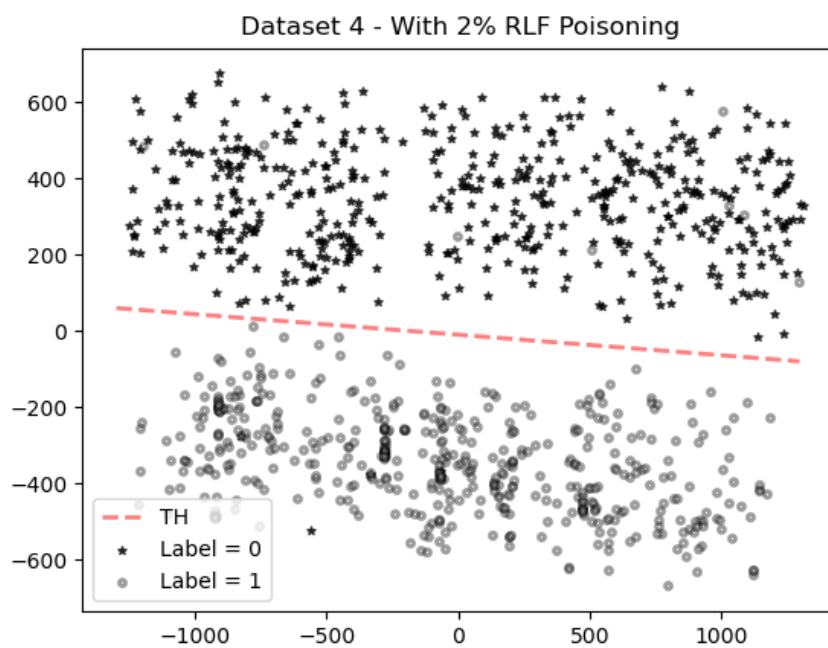


Figure 19 Run RF with 2% Poisoned Dataset 4

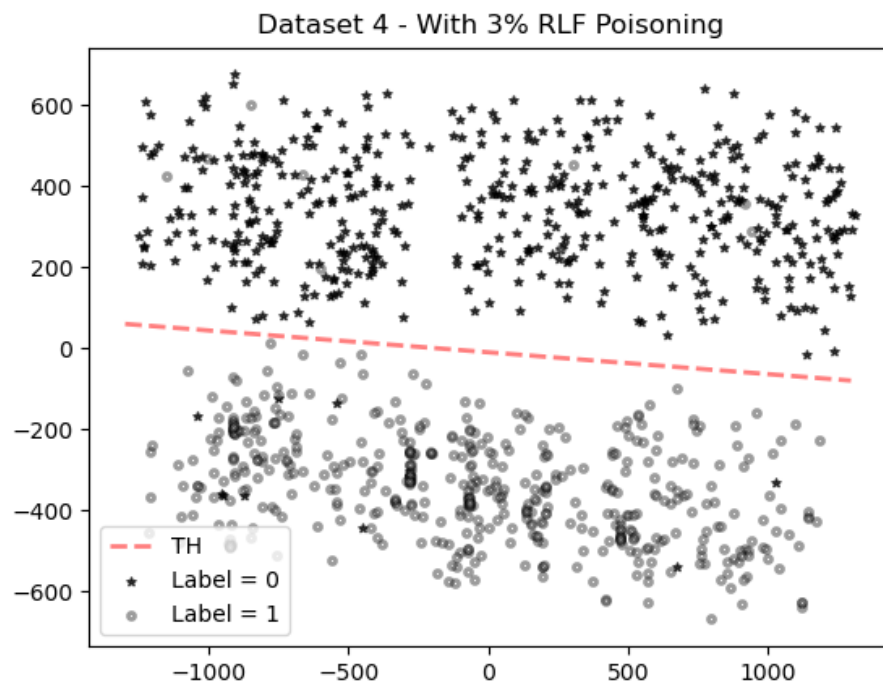


Figure 20 Run RF with 3% Poisoned Dataset 4

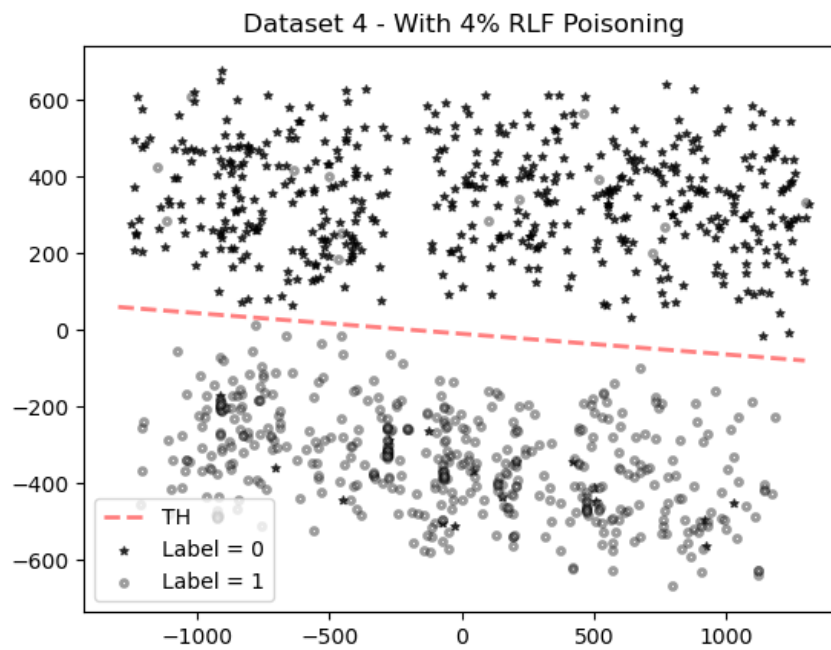


Figure 21 Run RF with 4% Poisoned Dataset 4

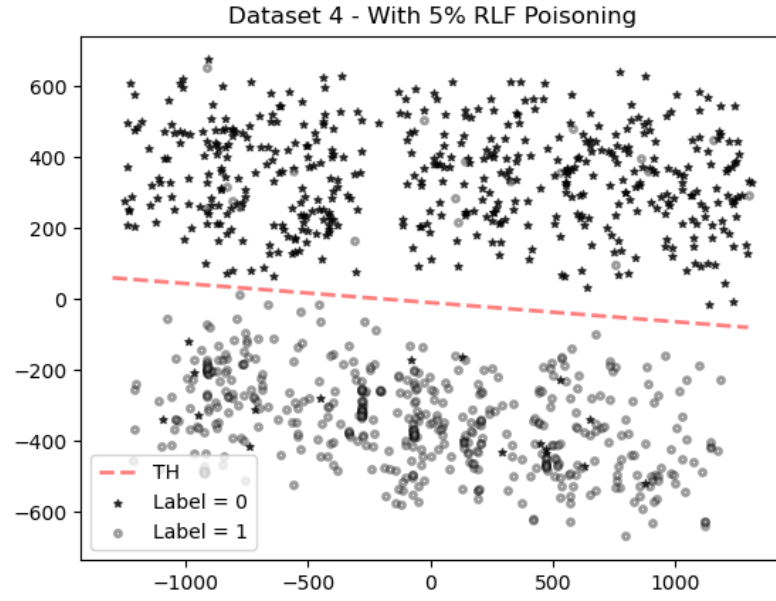


Figure 22 Run RF with 5% Poisoned Dataset 4

5.3 Apply Defense on ML Models

5.3.1 Find Optimal K Value

We assessed the efficacy of our defense mechanism against RLF attacks on RF and XGBoost classifiers. This defense mechanism is based on the K -NN approach that predicts true labels by identifying the K nearest labels from the original dataset D to each label from the untrusted dataset D' . Initially, our focus was on determining the optimal value of K for each dataset to minimize the error rate in our results using the original datasets. This mechanism depends on the availability and quality of a trusted dataset to identify the best value of K . Table 8 displays the optimal K value for each of the six original datasets in Table 3. The optimal K values is obtained through Algorithm 2 in the appendix, showcasing variations in K values across different datasets. For instance, Dataset 1 is assigned a K value of "33," while Dataset 2 is assigned a K value of "5."

Table 8

The Optimal Value of K for All Six Datasets

Original Dataset	K Value
Original Dataset#1	33
Original Dataset#2	5
Original Dataset#3	3
Original Dataset#4	3
Original Dataset#5	3
Original Dataset#6	3

5.3.1 Apply Defense on RF

Employing the obtained K values, we implemented our defense mechanism to detect poisoned labels, trigger an alarm for each identified poisoned label, and recover the value of the poisoned label. The execution of the defense mechanism on the entire dataset resulted in a fully recovered dataset from the applied RLF attack. The achieved accuracy rates after applying the K -NN defense mechanism are demostated in Table 9.

Table 9

RF Accuracy Rate with K -NN Defense against RLF Attack

Utilized Dataset	Poison Rate	Training Acc.	Detected Flipped Labels
<i>Dataset#1</i>	2%	99.87%	17
	3%	99.87%	23
	4%	100.00%	31
	5%	100.00%	40
<i>Dataset#2</i>	2%	99.87%	18
	3%	100.00%	24
	4%	100.00%	34
	5%	99.87%	40
<i>Dataset#3</i>	2%	100.00%	18
	3%	100.00%	24
	4%	100.00%	31
	5%	100.00%	40
<i>Dataset#4</i>	2%	100.00%	16

	3%	100.00%	20
	4%	100.00%	30
	5%	100.00%	38
	2%	100.00%	18
<i>Dataset#5</i>	3%	100.00%	26
	4%	99.89%	34
	5%	100.00%	42
	2%	100.00%	16
<i>Dataset#6</i>	3%	100.00%	24
	4%	100.00%	32
	5%	99.87%	38

Based on the experimental results presented in Table 9, our defense mechanism successfully demonstrated a high success rate in recovering the majority of the poisoned labels, leading to an improvement in training accuracy. For example, with a 5% poison rate, the training accuracy rate of Dataset 3 increased from 95.19% to 100%, recovering 40 detected poisoned labels, while Dataset 2 exhibited an increase in training accuracy rate from 95.18% to 99.87% by recovering 40 detected poisoned labels. This underscores the efficacy of our proposed defense mechanism in identifying poisoned labels and flipping them back to their true label. It is noteworthy that achieving 100% accuracy in most cases might seem impractical, but this is attributed to the repeated training of the RF model on the same dataset.

To further analyze the effectiveness of our proposed method, we generated plots illustrating the recovered datasets subsequent to the implementation of our K -NN defense mechanism. Through a comparison between the plots of the poisoned dataset and the recovered dataset, the effectiveness of our approach becomes visually apparent. Figures 27 to 43 showcase recovered Dataset 1, 2, 3 & 4, initially manipulated by 2-5% of its labels, subsequently recovered by our K -NN defense mechanism. In the following plots, benign and malicious labels are accurately split with respect to the red decision margin. Comparing between the recovered Dataset 1 in Figure 30 with the original Dataset 1 in Figure 14 provides further analysis and reveals a minor dissimilarity, in specific, a label on the decision margin is denoted as

"0" in Figure 14, whereas the same label is denoted as "1" in Figure 30. To enhance clarity in this. We included Figure 31 to illustrate the disparity between both figures.

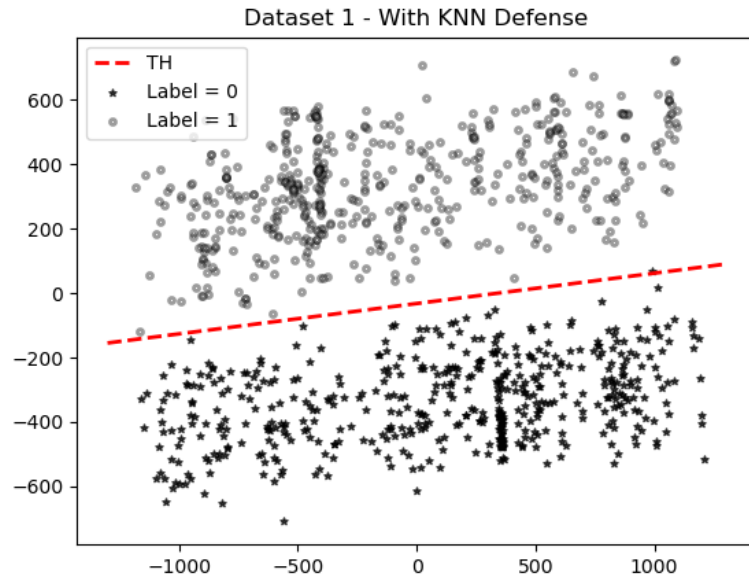


Figure 23 Recovered Dataset 1 from 2% poisoned samples

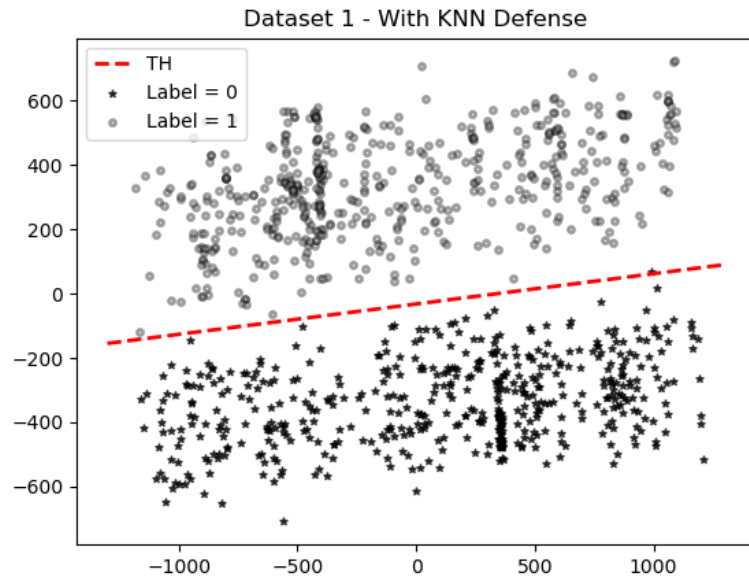


Figure 24 Recovered Dataset 1 from 3% poisoned samples

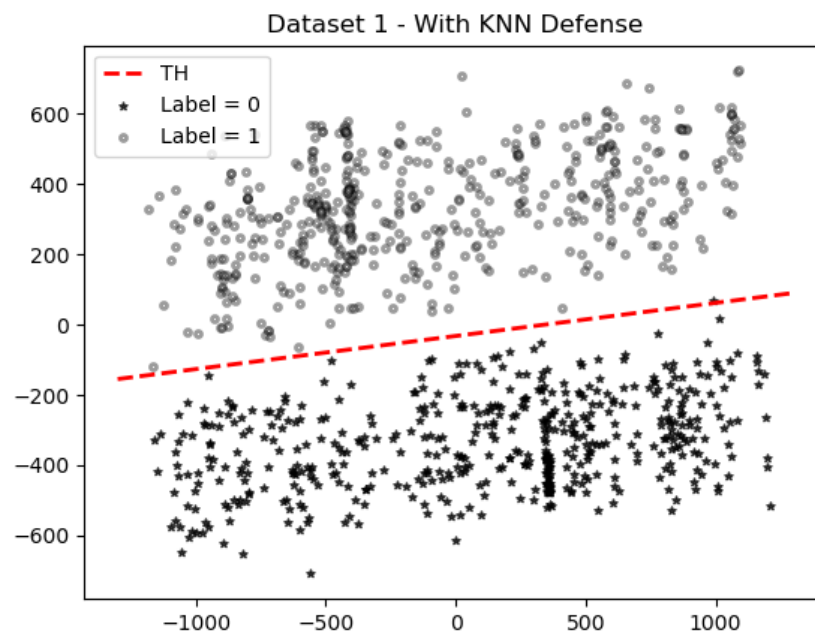


Figure 25 Recovered Dataset 1 from 4% poisoned samples

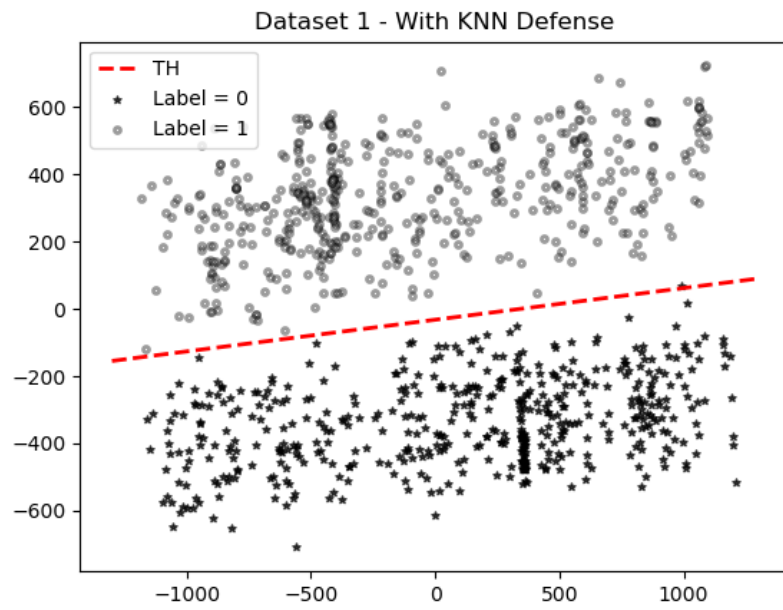


Figure 26 Recovered Dataset 1 from 5% poisoned samples

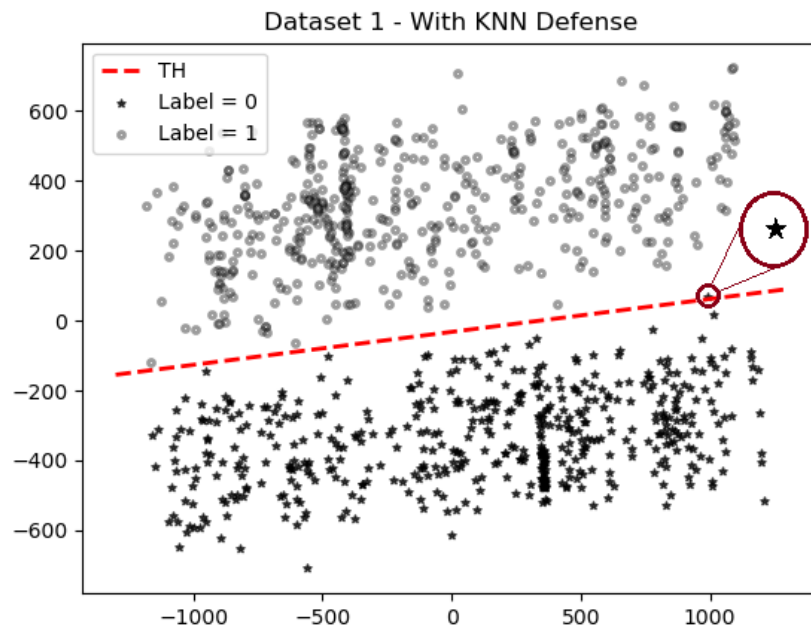


Figure 27 A duplicate of Figure 27 to highlight the difference between recovered & original dataset

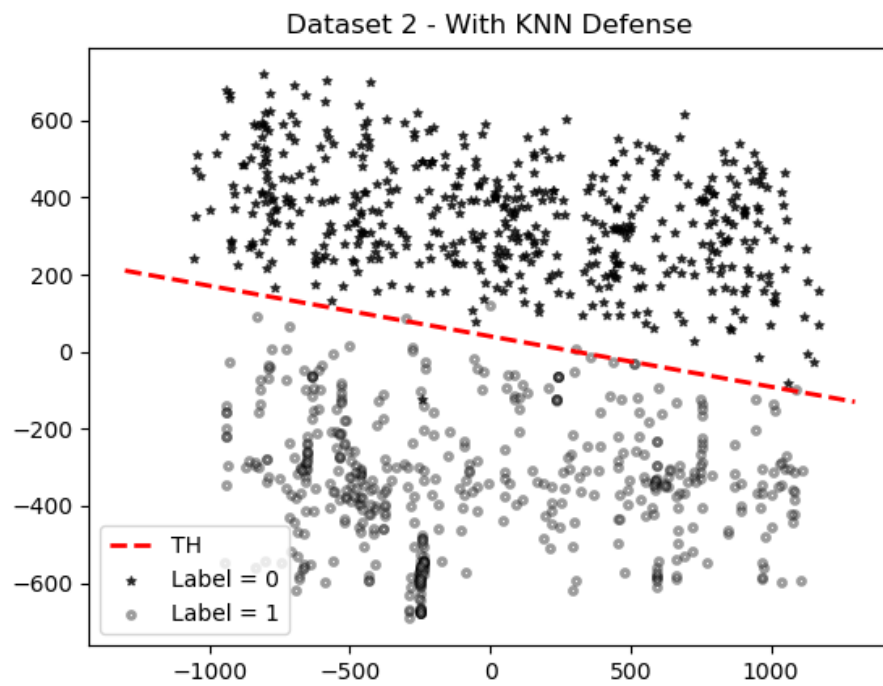


Figure 28 Recovered Dataset 2 from 2% poisoned samples

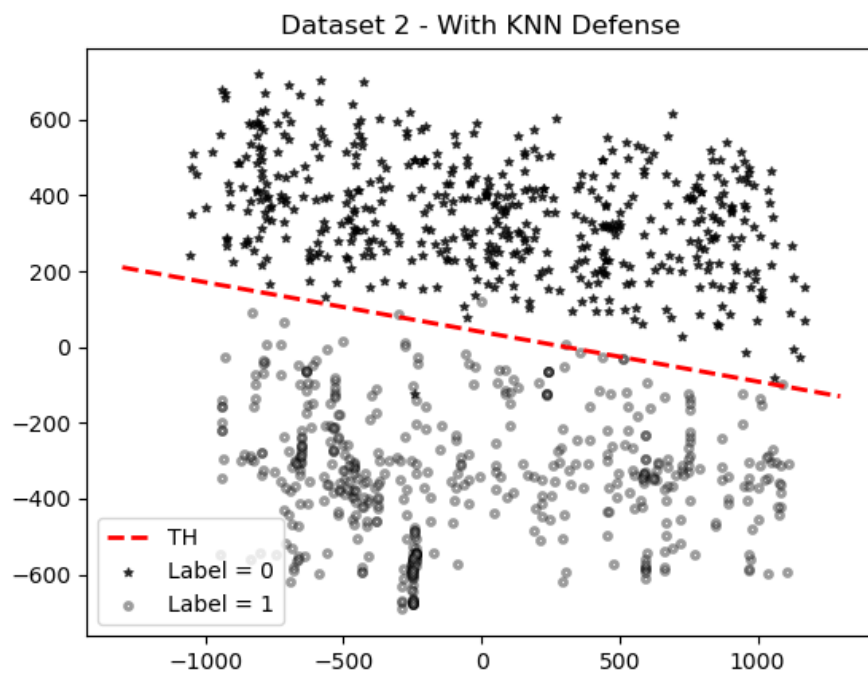


Figure 29 Recovered Dataset 2 from 3% poisoned samples

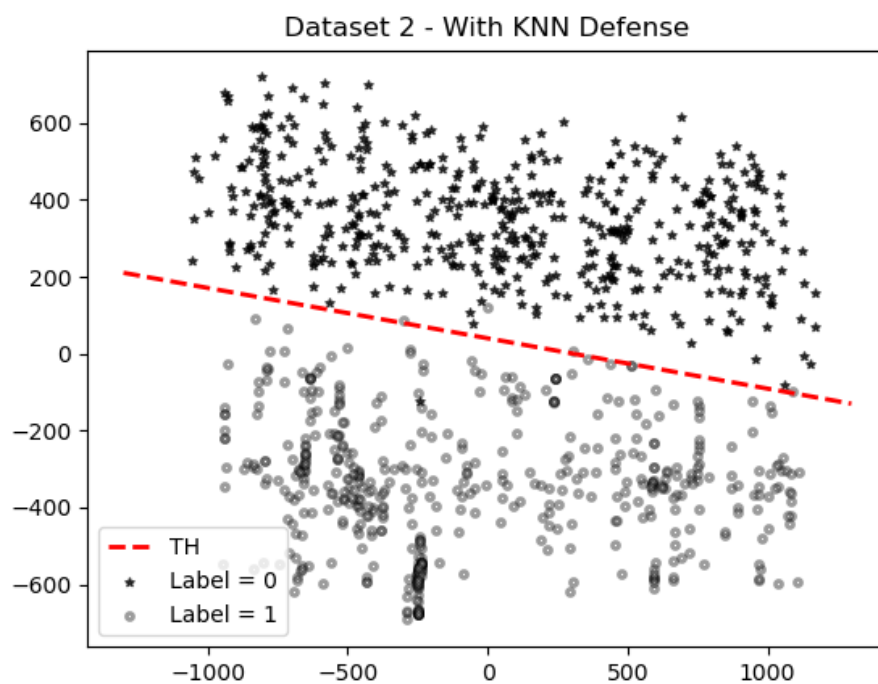


Figure 30 Recovered Dataset 2 from 4% poisoned samples

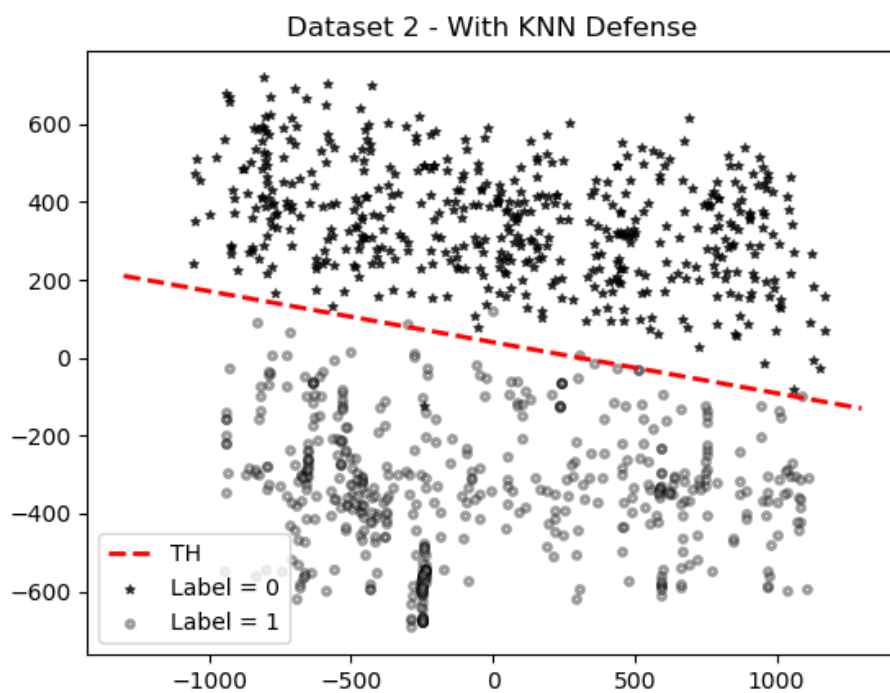


Figure 31 Recovered Dataset 2 from 5% poisoned samples

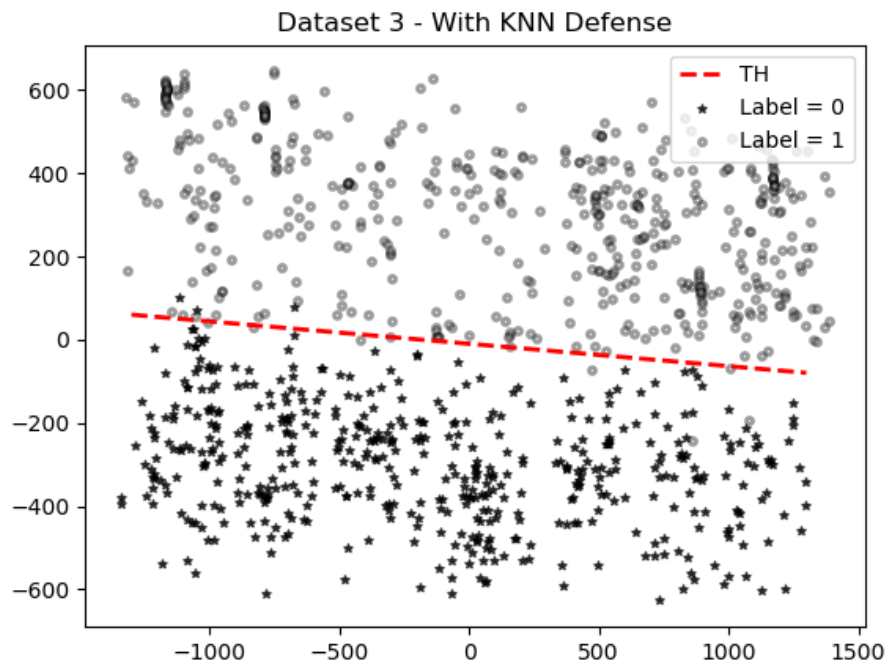


Figure 32 Recovered Dataset 3 from 2% poisoned samples

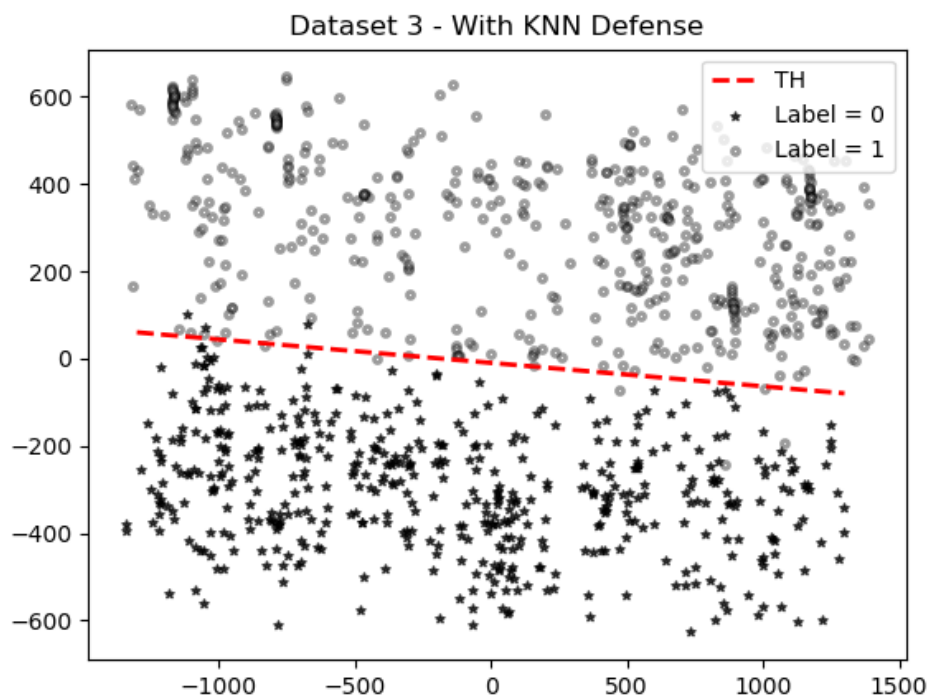


Figure 33 Recovered Dataset 3 from 3% poisoned samples

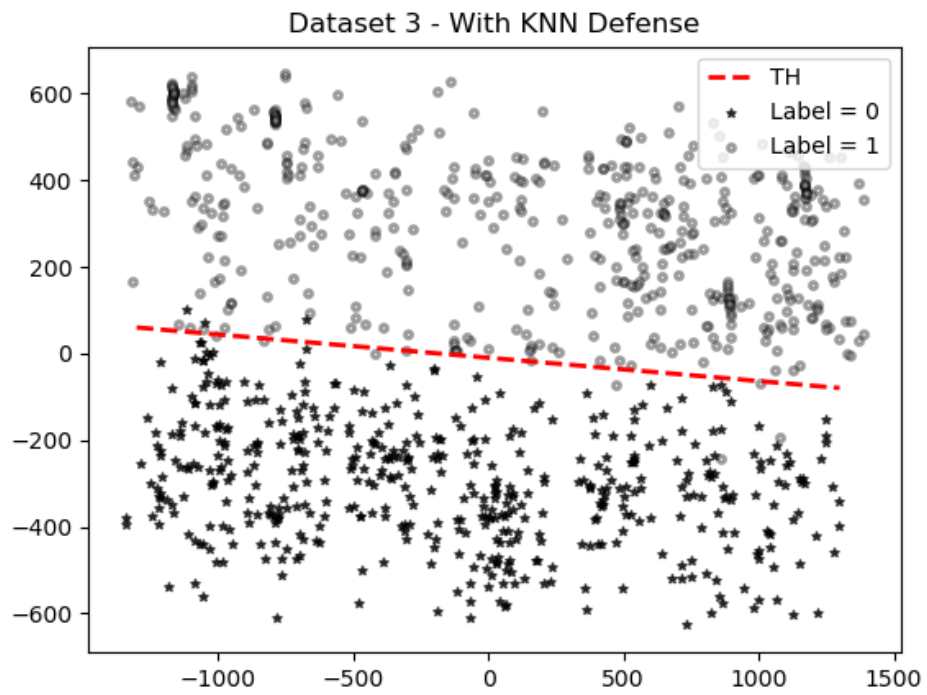


Figure 34 Recovered Dataset 3 from 4% poisoned samples

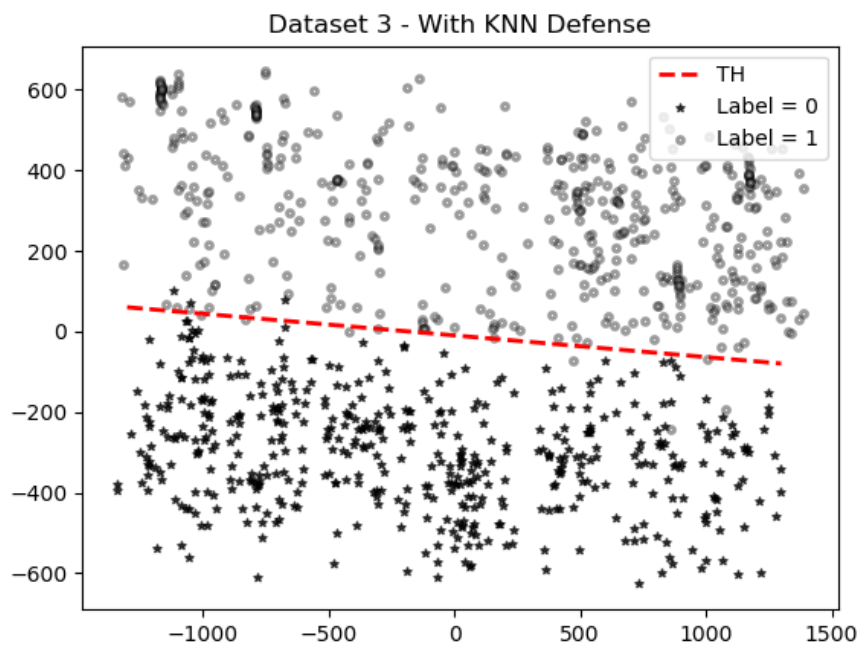


Figure 35 Recovered Dataset 3 from 5% poisoned samples

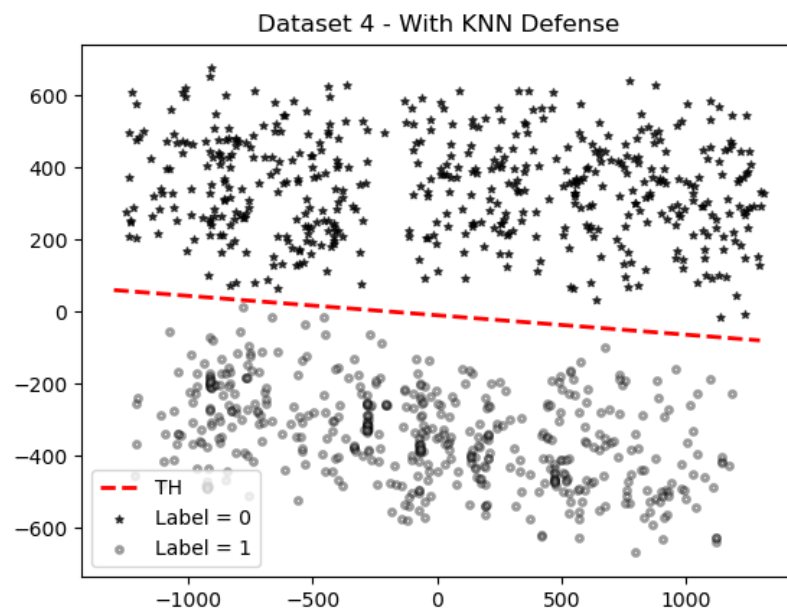


Figure 36 Recovered Dataset 4 from 2% poisoned samples

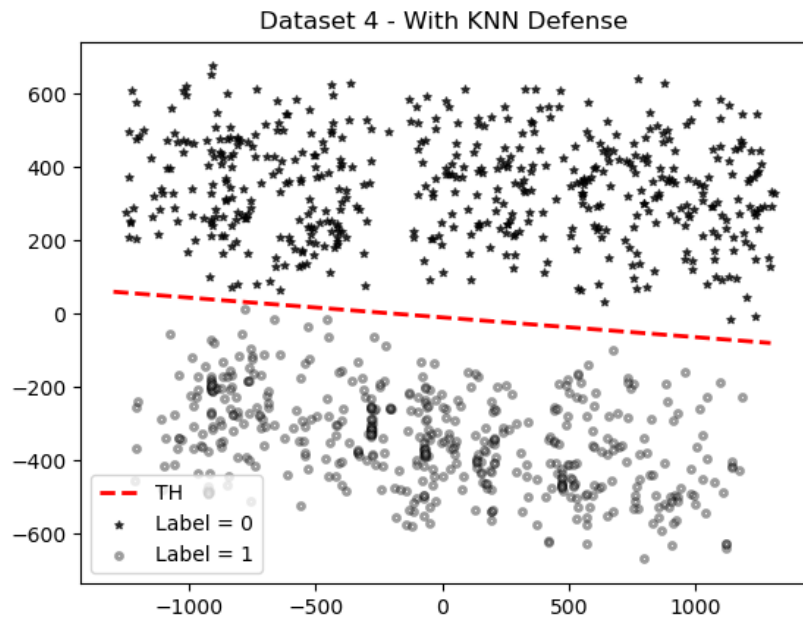


Figure 37 Recovered Dataset 4 from 3% poisoned samples

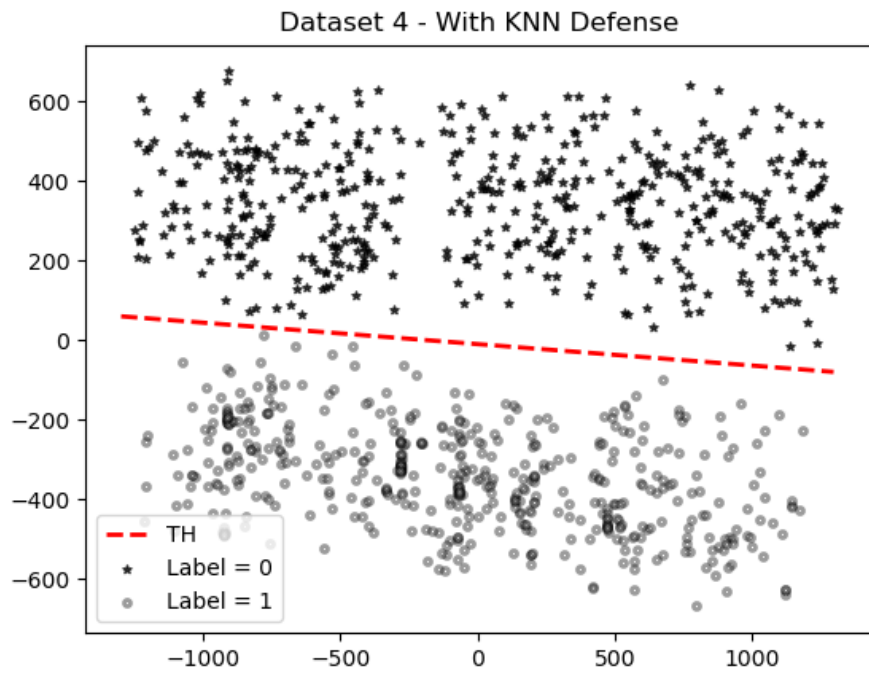


Figure 38 Recovered Dataset 4 from 4% poisoned samples

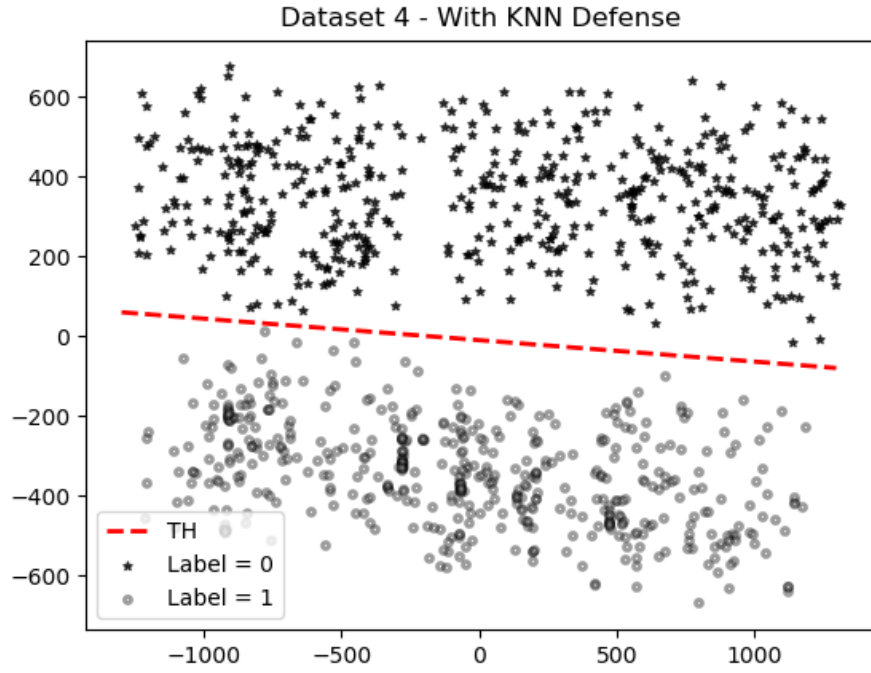


Figure 39 Recovered Dataset 4 from 5% poisoned samples

Before we finalize our experiment, we applied the proposed K -NN defense mechanism against RLF attack on XGBoost classifiers. The obtained results are outlined in Table 10. It is clearly observed that the defense method improved the training accuracy rate and detected the poisoned labels providing a fully recovered dataset.

Table 10

XGBoost Accuracy Rate with K -NN Defense against RLF Attack

Utilized Dataset	Poison Rate	Training Acc.	Detected Flipped Labels
<i>Dataset#1</i>	2%	99.84%	17
	3%	99.84%	23
	4%	99.84%	31
	5%	99.84%	41
<i>Dataset#2</i>	2%	99.68%	18
	3%	99.68%	22
	4%	99.68%	32

	5%	99.68%	42
	2%	99.36%	15
<i>Dataset#3</i>	3%	99.36%	23
	4%	99.36%	33
	5%	99.36%	43
	2%	100.00%	16
<i>Dataset#4</i>	3%	100.00%	24
	4%	100.00%	30
	5%	100.00%	40
	2%	98.41%	18
<i>Dataset#5</i>	3%	98.41%	24
	4%	98.41%	32
	5%	98.41%	38
	2%	99.84%	16
<i>Dataset#6</i>	3%	99.84%	24
	4%	99.84%	32
	5%	99.84%	36

To summarize our experimental results, we computed the RF and XGBoost models on the original six datasets to scrutinize its inherent behavior. This served as a baseline for measuring the impact of LF attacks and evaluating the efficacy of our proposed defense approach. Our observations highlighted the potential vulnerability of RF and XGBoost models to label poisoning, manifesting in a marginal accuracy degradation within the range of 95-98%, which is typically considered negligible. Ultimately, the implementation of the K-NN defense method showcased its effectiveness in mitigating the adverse effects of label poisoning, manifesting in significant enhancements in both model accuracy and robustness, reaching impressive levels of 99.87-100% with the recovered datasets. It is imperative to underscore the critical role of having a trustworthy dataset to successfully implement our defense mechanism.

Chapter 6

Conclusion and Future Work

Within this study, we illuminate several pivotal facets of RLF attacks targeting malicious URL detectors and explore potential defense strategies based on our analytical findings. Initially, a LF attack is executed by randomly flipping benign and malicious labels at a modest poisoning rate of 2-5%. Our study reveals that corrupting samples within the training dataset during the training phase, achieving an ASR between 50-65% within a 2-5% poisoning rate, renders this attack a viable choice for an attacker aiming to impact model performance significantly. It's noteworthy that the LF attack effectively deceived the detector, assuming the role of a ghost. Consequently, the implementation of a robust defense model becomes imperative to counteract the repercussions of LF attacks and restore the integrity of poisoned datasets. Our defense mechanism demonstrates efficacy by successfully identifying poisoned labels, predicting true labels, and enhancing model accuracy up to 100%.

Our thesis holds the potential to serve as a valuable instrument for evaluating the resilience of RF and XGBoost models and enhancing their efficacy in countering malicious URLs. Future research endeavors will need to address various aspects, including recognizing the limitation of fortifying the robustness of the proposed defense strategy in the absence of a trusted dataset. Additionally, it should explore other forms of backdoor attacks in URL detectors utilizing the RF and XGBoost classifier to devise potential mechanisms for recognizing and defending against such attacks. The primary goal of these mechanisms would be to detect the presence of backdoors within the training dataset through a comprehensive analysis of feature distribution in the samples. Consequently, further investigations are poised to play a pivotal role in advancing not only our study but also the broader realm of ML security.

REFERENCES

- Advertisement, M. (2023). Retrieved from <https://learn.microsoft.com/en-us/security/engineering/securing-artificial-intelligence-machine-learning>
- Ahmed, I. M. (2021). Threats on machine learning technique by data poisoning attack: A survey. In I. M. Ahmed, *Advances in Cyber Security: Third International Conference, ACeS 2021, Penang, Malaysia, August 24--25, 2021, Revised Selected Papers 3* (pp. 586--600). Springer.
- Alcal-Fdez, J. a. (2011). Keel data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework. *Journal of Multiple-Valued Logic and Soft Computing*, 255--287.
- Anisetti, M. a. (2022). On the robustness of ensemble-based machine learning against data poisoning. *arXiv preprint arXiv:2209.14013*.
- Arp, D. a. (2014). Drebin: Effective and explainable detection of android malware in your pocket. *Ndss*, 23--26.
- Carlini, N. a. (2017). Adversarial examples are not easily detected: Bypassing ten detection methods. In *Proceedings of the 10th ACM workshop on artificial intelligence and security* (pp. 3-14).
- Chan, P. P. (2021). Causative label flip attack detection with data complexity measures. *International Journal of Machine Learning and Cybernetics*, 12.
- Chang, J. Y. (2020). Data Poisoning Attack on Random Forest Classification Model. *Proc. of SMA 2020*.
- Chapman, D. J. (1994). *Musk (Version 2)*. Retrieved from UCI Machine Learning Repository: DOI: <https://doi.org/10.24432/C51608>.
- Cheng, N. a. (2021). Data sanitization against label flipping attacks using AdaBoost-based semi-supervised learning technology. *Soft Computing*, 25.
- Cisco. (2023). *Malware Detection and Protection*. Retrieved from Cisco Umbrella: <https://umbrella.cisco.com/trends-threats/malware-detection-and-protection>

- Contagio, D. (2023). Retrieved from Contagio: <http://contagiomindump.blogspot.com/>.
- Deng, L. (2012). The mnist database of handwritten digit images for machine learning research [best of the web]. *IEEE signal processing magazine*, 29.
- Derrac, J. a. (2015). Keel data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework. *J. Mult. Valued Log. Soft Comput*, 255--287.
- Dunn, C. a. (2020). Robustness evaluations of sustainable machine learning models against data poisoning attacks in the internet of things. *Sustainability*, 12.
- E.Nowroozi. (2023). *Ensemble trees: A defense mechanism against label flipping in malicious URL detection*. Retrieved from Ensemble trees: A defense mechanism against label flipping in malicious URL detection: https://github.com/ehsannowroozi/LF_Attack_Defense_URL
- Gardiner, J. a. (2016). On the security of machine learning in malware c&c detection: A survey. *ACM Computing Surveys (CSUR)*, 49.
- Gilmer, J. a. (2018). Motivating the rules of the game for adversarial example research. *arXiv preprint arXiv:1807.06732*.
- Goldblum, M. a. (2022). Dataset security for machine learning: Data poisoning, backdoor attacks, and defenses. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45.
- Google, A. (2023). Retrieved from <https://ai.google/responsibility/responsible-ai-practices/>
- He, K. a. (2023). Adversarial machine learning for network intrusion detection systems: a comprehensive survey. *IEEE Communications Surveys & Tutorials*.
- Hopkins, M. R. (1999). *Spambase*. Retrieved from UCI Machine Learning Repository: <https://doi.org/10.24432/C53G6X>
- Jebreel, N. M.-F.-J. (2022). Defending against the label-flipping attack in federated learning. *arXiv preprint arXiv:2207.01982*.

- Kuipers, M. &. (2022). Journey of Artificial Intelligence. *Wireless Personal Communications*, 16.
- Kumar, R. S. (2020). Adversarial machine learning-industry perspectives. In *2020 IEEE Security and Privacy Workshops (SPW)* (pp. 69-75).
- Lin, J. a. (2021). ML attack models: Adversarial attacks and data poisoning attacks. arXiv 2021. *arXiv preprint arXiv:2112.02797*.
- Liu, X. a. (2021). Privacy-enhanced federated learning against poisoning adversaries. *IEEE*, 16.
- Mcafee. (2023). *Mcafee webadvisor*. Retrieved from Mcafee webadvisor: <https://www.mcafee.com/en-us/safe-browser/mcafee-webadvisor.html>
- Moisejevs, I. (2019). Evasion attacks on machine learning (or “adversarial examples”). *Towards Data Science*, accessed July.
- Nowroozi, E. a. (2022). Demystifying the transferability of adversarial attacks in computer networks. *IEEE Transactions on Network and Service Management*, 19.
- Nowroozi, E. a. (2023). Employing Deep Ensemble Learning for Improving the Security of Computer Networks against Adversarial Attacks. *IEEE Transactions on Network and Service Management*.
- Paudice, A. a.-G. (2018). Detection of adversarial training examples in poisoning attacks through anomaly detection. *arXiv preprint arXiv:1802.03041*.
- Paudice, A. a.-G. (2019). Label sanitization against label flipping poisoning attacks. In *ECML PKDD 2018 Workshops: Nemesis 2018, UrbReas 2018, SoGood 2018, IWAISe 2018, and Green Data Mining 2018, Dublin, Ireland, September 10-14, 2018, Proceedings 18* (pp. 5--15). Springer.
- Ramirez, M. A.-J.-Y.-S. (2022). New data poison attacks on machine learning classifiers for mobile exfiltration. *arXiv preprint arXiv:2210.11592*.
- Reyes-Ortiz, J. A. (2012). *Human Activity Recognition Using Smartphones*. Retrieved from UCI Machine Learning Repository: {DOI}: <https://doi.org/10.24432/C54S4K>

- Sankaran, M. A. (2021). Detection Of Malicious Urls Using Machine Learning Techniques. *Int. J. of Aquatic Science*, 12.
- Shahid, A. R. (2022). Label flipping data poisoning attack against wearable human activity recognition system. In *2022 IEEE Symposium Series on Computational Intelligence (SSCI)* (pp. 908--914). IEEE.
- Shapiro, A. (1989). *Chess (King-Rook vs. King-Pawn)*. Retrieved from UCI Machine Learning Repository: DOI: <https://doi.org/10.24432/C5DK5C>.
- Symantec. (2023). *Symantec-endpoint-protection*. Retrieved from Symantec: <https://techdocs.broadcom.com/us/en/symantec-security-software/endpoint-security-and-management/endpoint-protection/all/symantec-endpoint-protection-client-for-windows-help/enabling-network-traffic-redirection.html>
- Taheri, R. a. (2020). On defending against label flipping attacks on malware detection systems. *Neural Computing and Applications*, 32.
- Tavallali, P. a. (2022). Adversarial label-poisoning attacks and defense for general multi-class models based on synthetic reduced nearest neighbor. In *2022 IEEE International Conference on Image Processing (ICIP)* (pp. 3717--3722). IEEE.
- UCI, M. L. (2023). *UCI Machine Learning Repository*. (n.d.). Retrieved from <http://archive.ics.uci.edu/>.
- Wang, C. a. (2022). Poisoning attacks and countermeasures in intelligent networks: Status quo and prospects. *Digital Communications and Networks*, 8.
- Wolberg, W. (1992). *Breast Cancer Wisconsin (Original)*. Retrieved from UCI Machine Learning Repository.: DOI: <https://doi.org/10.24432/C5HP4Z>
- Wolberg, W. M. (1995). *Breast Cancer Wisconsin (Diagnostic)}*. Retrieved from UCI Machine Learning Repository: <https://doi.org/10.24432/C5DW2B>
- Yerlikaya, F. A. (2022). Data poisoning attacks against machine learning algorithms. *Expert Systems with Applications*, 208.
- Zhou, Y. a. (2012). Dissecting android malware: Characterization and evolution. In *2012 IEEE symposium on security and privacy* (pp. 95--109). IEEE.