



Avinashilingam Institute for Home Science and Higher Education for Women

Deemed to be University Estd. u/s 3 of UGC Act 1956, Category A by MHRD

Re-accredited with 'A++' Grade by NAAC.CGPA 3.65/4, Category I by UGC

Coimbatore-641 043, Tamil Nadu, India

**DETECTION AND MITIGATION OF MALICIOUS INSIDER THREATS
IN CLOUD ENVIRONMENT USING MACHINE LEARNING METHODS**

Centre for Machine Learning and Intelligence

AI project sanctioned under DST-CURIE-AI, Phase II 2021-2022

Center for Cyber Intelligence



Asha .S (20PHCSF005)

Research Scholar in Computer Science

Under the Supervision of

Dr. G. Padmavathi,

**Dean - School of Physical and Computational Sciences (PSCS) and Professor
Department of Computer Science**

Dr. D. Shanmugapriya

Head and Assistant Professor

Department of Information Technology

School of Physical and Computational Sciences (PSCS)

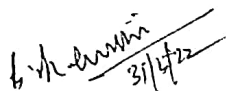
Avinashilingam Institute for Home Science and Higher Education for Women

MAY 2022

CERTIFICATE

CERTIFICATE

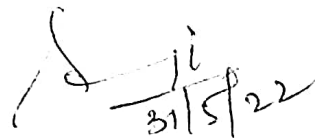
This is to certify that this project work entitled "DETECTION AND MITIGATION OF MALICIOUS INSIDER THREATS IN CLOUD ENVIRONMENT USING MACHINE LEARNING METHODS" done by the Research Scholar Asha. S (20PHCSF005) has been submitted to Avinashilingam Institute for Home science and Higher education for women, Coimbatore-43 in partial fulfillment of the project sanctioned under DST-CURIE-AI Phase II. Certified as a bonafied record of the work submitted for the project completion report.



(Dr. G. Padmavathi)

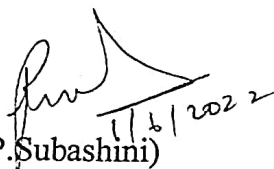
Signature of Principal Investigator

Dr. G.PADMAVATHI
M.Sc.,M.Phil.,Ph.D.,MISTE, MCSI.,
Dean, School of Physical Science and
Computational Sciences
Avinashilingam Institute for Home Science
and Higher Education for Women
(Deemed to be University)
Coimbatore - 641 043



(Dr.D.Shanmugapriya)

Signature of Co-principal Investigator



(Dr. P. Subashini)

Signature of Project Coordinator

DECLARATION

DECLARATION

I hereby declare that the project entitled "DETECTION AND MITIGATION OF MALICIOUS INSIDER THREATS IN CLOUD ENVIRONMENT USING MACHINE LEARNING METHODS" is a record of the original work done by Asha. S (20PHCSF005) under the guidance of Dr. G. Padmavathi and Dr. D. Shanmugapriya and this project work has not formed the basis for any Degree/Diploma/Associates.

PLACE: COIMBATORE

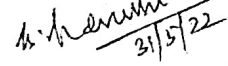
DATE: 31/5/2022

Asha. S

(Asha. S)

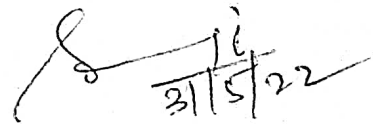
Signature of Research Scholar

Dr. G. PADMAVATHI
M.Sc., M.Phil., Ph.D., MISTE., MCSI.,
Dean, School of Physical Science and
Computational Sciences
Avinashilingam Institute for Home Science
and Higher Education for Women
(Deemed to be University)
Coimbatore - 641 043


31/5/22

Countersigned By

Dr. G. Padmavathi (Principal Investigator),
Dean - School of Physical and Computational Sciences and Professor,
Department of Computer Science
School of Physical and Computational Sciences (PSCS)


31/5/22

Dr. D. Shanmugapriya (Co-Principal Investigator)
Head and Assistant Professor
Department of Information Technology
School of Physical and Computational Sciences (PSCS)

ACKNOWLEDGEMENT

ACKNOWLEDGEMENT

I would like to express my sincere thanks to **God** Almighty, for his constant love and grace that he has shown upon me, which kept me in good health, and sound mind without which my project would not have reached a successful end.

I would like to express my deep sense of reverential gratitude and sincere thanks to **Dr.S.P.Thyagarajan**, Chancellor, Avinashilingam Institute of Home Science and Higher Education for Women, Coimbatore, for the opportunity given to me for undertaking this study and for providing all the needed facilities during my study.

I owe my great deal of gratitude to **Dr.V.Bharathi Harishankar, Ph.D., FRSA Vice-Chancellor**, Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore, for extending all resources that facilitated the conduct of the present study.

I owe my great deal of gratitude to **Dr.PremavathyVijayan M.Sc., M.Ed., Dip. Spl. Edn., M.Phil., Ph.D., Former Vice Chancellor**, Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore, for extending all resources that facilitated the smooth conduct of the project study.

I express my gratitude to **Dr.S.Kowsalya, Registrar, M.Sc., M.Phil., Ph.D.** Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore, for providing all facilities necessary for the study.

I would express my boundless thanks to **Dr. G. Padmavathi, M.Sc., M.Phil., Ph.D., and Dean**, School of Physical Sciences & Computational Sciences and Principal Investigator for Center for Cyber Intelligence, Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore, for granting the facility required.

I wish to place on record my deep sense of gratitude to **Dr. Vasantha Kalyani David, M.Sc., M.Phil(Mathematics), M.Phil(CS), Ph.D., Professor and Head**, Department of Computer Science for support and encouragement to complete the project.

I am grateful to the project coordinator **Dr. (Mrs.) P.Subashini, MCA., M.Phil., M.Sc.(Applied Psychology), Ph.D., Professor of Computer Science**, who was instrumental in granting me the facilities required for doing a project.

I am grateful to the Co-principal investigator **Dr. D. Shanmugapriya, Assistant Professor and Head**, Department of Information Technology Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore, for providing all facilities necessary for the study.

I also like to extend my gratitude to **Ms. A. Roshni, M.Sc., Technical Assistant of CCI**, Department of Computer Science, project Guidelines CCI always supported me and nurtured me with valuable advice and profound belief in my work and abilities.

It was a great feeling to finally complete my project with the full support of the **Center for Cyber Intelligence**. Which I had been working on for so long. Thank you so much for always paying close attention to my work and recognizing my accomplishments.

Finally, yet importantly, I would like to thank my **parents, family members, and friends** for their kind inspiration, support, encouragement, blessings, and prayers, which were instrumental in the successful completion of the project.

I have great pleasure in expressing my deep sense of gratitude to all other teaching and non-teaching staff members of the Department of Computer Science, who stood behind the screen for the completion of the project. I would extend my hearty thanks to one and all that helped me directly or indirectly with the successful completion of my project.

ABSTRACT

ABSTRACT

Cloud computing is a framework that provides infrastructure, platform and software as a service to a wide range of users at a metered cost. It is more beneficial to the end user but it is prone to numerous security threats. Some security threats in the cloud can be handled using a built-in security mechanism. However, it fails to handle the more destructible passive attack such as malicious insider threats. Malicious Insider may be a current / past employee of an organization who can steal the confidential data leading to data breaches. So, it is important to detect and mitigate the malicious insider from the network to enhance the security of the cloud.

The aim of the project is to propose the techniques for detecting and mitigating malicious insider. One way of detecting the malicious insider threat is by applying anomaly detection techniques. Since the class imbalance problem arises in these techniques, the data level sampling methods are recommended. The performance of different types of undersampling and oversampling techniques are evaluated based on the performance metrics such as precision, recall, f-score and accuracy. The best performing sampling technique is used in anomaly detection technique for further detection. In this project, supervised machine learning based anomaly detection using one-class support vector machine (OCSVM) with variants of sampling techniques are implemented for detecting the attack and Multifactor Authentication with keystroke based biometric authentication and OTP based secondary authentication is implemented to mitigate the malicious insider threat. In detection phase, the true detection rate is of 100% and false detection rate is of 0% to detect all the malicious activity in top 50%. The user who performs the malicious activity is undergone the mitigation phase. In mitigation phase, the biometric authentication verifies the user as genuine if the value of EER is low, the user is labelled as genuine and is subjected to OTP-based secondary authentication. The user who performs successful OTP verification is considered as genuine and has gained access to security system.

Keywords: Anomaly Detection; Biometric Authentication; Cyber Attack; Insider Threat; Multifactor Authentication;

CONTENTS

TABLE OF CONTENTS

CHAPTER NO.	DESCRIPTIONS	PAGE.NO
1.	INTRODUCTION	1
	1.1 CLOUD COMPUTING	1
	1.2 TYPES OF SECURITY ATTACKS IN CLOUD	1
	1.2.1 ACTIVE ATTACK	2
	1.2.2 PASSIVE ATTACK	2
	1.3 INSIDER THREAT	4
	1.4 TYPES OF INSIDER THREAT	4
	1.5 WHY MI DETECTION AND MITIGATION NEEDED?	5
	1.6 CHARACTERISTICS OF INSIDER THREAT	6
	1.7 APPLICATIONS OF DETECTING MALICIOUS INSIDER THREAT	6
	1.8 APPLICATIONS OF MITIGATING MALICIOUS INSIDER THREAT	7
	1.9 PROBLEM DEFINITION	8
	1.10 OBJECTIVES	8
	1.11 ORGANIZATION OF THIS REPORT	8
SUMMARY	8	
2.	LITERATURE REVIEW	9
	2.1 STUDY OF CLASS IMBALANCE PROBLEM	9
	2.2 STUDY OF MALICIOUS INSIDER THREAT DETECTION	11
	2.3 STUDY OF BIOMETRIC AUTHENTICATION	15
	2.4 OBSERVATIONS FROM LITERATURE	22
	SUMMARY	22
3.	METHODOLOGY	23
	3.1 PHASE I: DETECTION PHASE	24
	3.1.1 SUB-PHASE I: DATASET	24

	3.1.2 SUB-PHASE II: DATA PRE-PROCESSING	26
	3.1.3 SUB-PHASE III: SUPERVISED LEARNING-BASED ANOMALY DETECTION	30
	3.2 RESULTS AND DISCUSSION IN PHASE I	34
	3.2.1 PERFORMANCE METRICS	34
	3.2.2 EXPERIMENTAL ANALYSIS	35
	3.3 PHASE II: MITIGATION PHASE	42
	3.3.1 SUB-PHASE I: DATA ACQUISITION	42
	3.3.2 SUB-PHASE II: BIOMETRIC AUTHENTICATION	43
	3.3.3 SUB-PHASE III: SECONDARY AUTHENTICATION	44
	3.4 RESULTS AND DISCUSSION IN PHASE II	47
	3.4.1 PERFORMANCE METRICS	47
	3.4.2 EXPERIMENTAL ANALYSIS	48
	SUMMARY	52
4	CONCLUSION	53
5	FUTURE ENHANCEMENT	54
	ACKNOWLEDGEMENT	55
	REFERENCES	56
	LIST OF PUBLICATIONS	62

LIST OF FIGURES

Figure no.	Descriptions	Page. no
1.1	Security attacks in cloud environment	1
1.2	Classification of Passive Security Attacks in Cloud	3
1.3	Classification of Insider Threat in Cloud Environment	4
3.1	Proposed Framework	23
3.2	Overview of malicious insider threat detection phase	24
3.3	Balanced data of Genuine users and malicious user	38
3.4	Activity count of genuine user and malicious user	39
3.5	Malicious insider based on employee activity	40
3.6	Malicious insider detected in an organization	41
3.7	The malicious activity of malicious employees in an organization	41
3.8	Overview of malicious insider threat mitigation phase	42
3.9	Overview of biometric authentication using keystroke dynamics	43
3.10	Secondary Authentication using OTP	45
3.11	Received OTP verification in email	49
3.12	OTP in the verification screen	50
3.13	Result of OTP verification in the dialogue box	51

LIST OF TABLES

Table no.	Descriptions	Page. no
1.1	Active attacks and its impacts	2
1.2	Passive attacks and its impacts	3
2.1	Study of Various Sampling Techniques	9
2.2	Study of variants of machine learning-based anomaly detection	13
2.3	Study of various biometric authentication using keystroke dynamics	17
3.1	Feature Details of Integrated data	26
3.2	Transformed data	27
3.3	Various sampling techniques and their working criteria	29
3.4	Performance metrics in phase I	34
3.5	Performance metrics of eight sampling methods	36
3.6	Comparison of SVM Classifier Performance using imbalanced and balanced data	37
3.7	Simulation parameters of OCSVM	38
3.8	Description of top X% of malicious activity	39
3.9	The activity of Malicious employees in top 50%	40
3.10	Performance metrics used in Phase II	47
3.11	Simulation parameters of GMM	48

INTRODUCTION

CHAPTER 1

INTRODUCTION

This chapter elaborates the basics of cloud, the security threat in cloud environment; malicious insider threat is vulnerable, recent attacks in malicious insider threats, the importance for detection of malicious insider threat in cloud environment in detail.

1.1 CLOUD COMPUTING

The storage and access of data and computer services through the internet is referred to as cloud computing. It provides the on-demand computer resources such as servers, data storage, networking, databases, and so on. i.e., software as a service, infrastructure as a service and platform as a service. The basic goal of cloud computing is to provide the computer services to various number of users from anywhere at any time.

In the cloud, there are two key players: the client and the provider. The customer would request cloud resources from the cloud service provider in the cloud (CSP). After receiving a request from a cloud user, the cloud service provider will allocate resources to the cloud user.

1.2 TYPES OF SECURITY ATTACKS IN CLOUD

During data transfer between a cloud client and a cloud service provider, an attacker may attack the cloud and attempt to steal the data. Both the Cloud Client and the Cloud Service Provider are vulnerable to two different forms of attacks: passive and active. Figure 1.1 shows the security attacks in cloud environment.

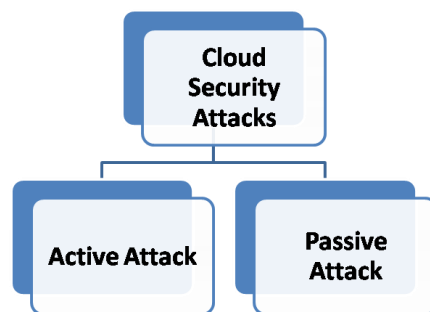


Figure 1.1: Security attacks in cloud environment

1.2.1 Active attack

In the cloud environment, an active attack is a type of security threat in which the intruder attempts to modify the cloud resources on the client side of the cloud. It is also known as cloud resource hijacking. According to GleekFlare, a US-based technical paper, the cloud computing has some security threats, and the impact of such vulnerabilities is reported. The categorization of active security attacks and its impacts in cloud environment is shown in table 1.1.

Table 1.1: Active attacks and its impacts [48]

Active attacks	Impacts
Misconfiguration and Insufficient Change Control	Data breach, modified resource and service interruption
Account Hijacking of cloud accounts	Confidential data leak
Weak Control Plane	Data Loss
Lack of Visibility in Cloud usage	Unauthorized access
Exploitation and Despicable Use of Cloud Services	Unauthorized access
Cyber-attacks	Data breach
Denial of Service Attacks	Data Loss

1.2.2 Passive attack

A passive attack is a type of security threat in which intruders attempt to take control of a network in order to spy on or gain cloud resources without changing any data on the client side of the cloud. It refers to a cloud resource eavesdropping approach in a cloud network. The categorization of Passive Security Attacks in a Cloud Environment is shown in figure 1.2:

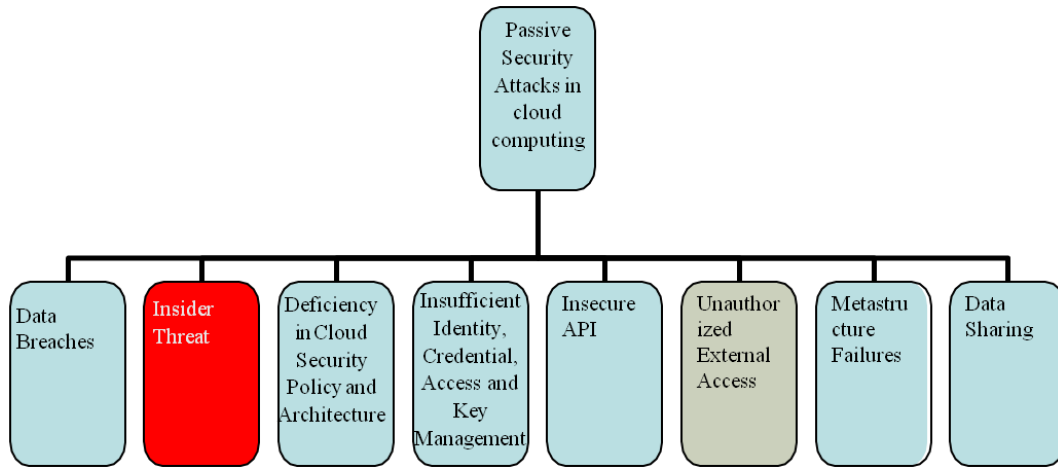


Figure 1.2: Classification of Passive Security Attacks in Cloud

The following table 1.2 describes the passive attacks and its impacts:

Table 1.2: Passive attacks and its impacts [48]

Passive Attacks	Impacts
Data Breaches	Loss of trust and damage business reputation
Malicious Insider	Damage business reputation and data leak
Deficiency in Cloud Security Policy and Architecture	Data breach
Insufficient Identity, Credential, Access and Key Management	Loss of Confidentiality, Security breach
Insecure Interfaces and APIs	Security breach
Unauthorized Access	Data loss and damage business reputation
External Data Sharing	Security breach and data loss

1.3 INSIDER THREAT

A cyber security threat that arises from within an organisation is referred to as an insider threat. It usually happens when a current or former employee, contractor, vendor, or partner who has legitimate user credentials abuses their access to the organization's networks, systems, and data. An insider threat might be carried either knowingly or unknowingly. Whatever the motivation, the ultimate effect is a breach of corporate system and data confidentiality, availability, and/or integrity. The majority of data breaches are caused by insider threats. Traditional cybersecurity plans, policies, processes, and systems frequently focus on external threats, leaving the company open to inside attacks. It's tough for security experts and apps to discern between routine and hazardous activities because the insider already has proper authorisation to data and systems.

1.4 TYPES OF INSIDER THREAT

The insider threat is categorized into three broad categories such as malicious insider, careless insider and a third party organization. Figure 1.3 depicts the classification of insider threat in cloud environment.



Figure 1.3: Classification of Insider Threat in Cloud Environment

- **Malicious Insider (MI)**
 - Malicious insider threats, sometimes known as turncloaks, are mainly concerned in espionage, fraud, intellectual property theft, and sabotage. They utilise their privileged access to steal information or impair systems for monetary, personal, and/or malevolent purposes. An employee who sells secret data to a rival or a dissatisfied former contractor who infects the organization's network with crippling malware are two examples.
- **Careless insider**
 - Inadvertent insider security threats happen all the time. Human mistake, poor decisions, unintended aiding and abetting, convenience, phishing (and other social engineering approaches), malware, and stolen passwords are all common causes of the careless insider in an organization. Unwittingly, the individual implicated exposes company systems to external assault is the extreme extent of the careless insider.
- **Third party organization**
 - An outsider who has gotten insider access to the organization's systems is known as a third party organization. They may impersonate a vendor, partner, contractor, or employee in order to get privileged authorization from a specified organization.

1.5 WHY MI DETECTION AND MITIGATION NEEDED?

The recent year experiences more number of malicious insider threat in an organization and made high reach in public due to its gigantic consequence. In 2021, the reputed organization experiences the malicious insider threat that lead to data leakage. Some of the cases are involving Marriott hotels and resorts, Elliott Greenleaf law firm, South Georgia Medical Center, Twitter, Ubiquiti Networks, Proofpoint, Saudi Aramco, Pfizer and UK Parliament. [1][2].

In 2022 cost of insider threat global report conducted by ponemon institute, the insider threats maximized in both occurrence and financial loss over past two years 2020. Ever since 2020, the insider threat incident is doubled. Furthermore, the minimum number cases reported for insider incident is one where the maximum number of cases is 46. From the reported insider threat incident in pass 12 month, 26% of cases are caused by malicious insider. It is analyzed that based

on the organization size, the incident of insider threat increases. The statistics shows that 52% of respondents are worried about the insider-driven data loss in cloud environment [3].

1.6 CHARACTERISTICS OF INSIDER THREAT

The characteristics listed below are used to detect and mitigate to possible insider threats:

- Unusual logins: Repeated login patterns must be recognised by security personnel. It's critical to maintain track of employee working hours across departments and to spot logins that happen at odd hours or from strange locations. Attempts to log in that fail should also raise concern.
- Unauthorized software was attempted for usage: Any effort by an employee to access a system that is dissimilar to their job function should trigger suspicion. An engineer attempting to access a CRM system or a salesman attempting to access a financial system, for example, should be detected and responded to by security personnel.
- Enhanced permission : Any attempt by workers to expand their own rights or by one employee to grant advantages to another should be thoroughly scrutinised. It's up to security teams to figure out who authorised the privilege escalation and if it is reasonable.
- Data infiltration or Exfiltration: Any employee who downloads or uploads huge amounts of data should be investigated by security. Users might be exfiltrating business data by uploading many GB of data to a cloud service or downloading files from a company server to a local device.
- Abnormal behaviour of employee: The change in employee behaviour, particularly in crucial jobs like as finance and IT administration, requires further insight. Employees that engage in antisocial behaviour, challenge superiors, are frequently absent from work, or work longer than normal should be investigated by security or HR.

1.7 APPLICATIONS OF DETECTING MALICIOUS INSIDER THREAT

The existing applications for detection of the malicious insider threat are explained below:

- Data Loss Prevention (DLP):
 - It identifies the possible incidents of illegal access to data, as well as efforts to exfiltrate or delete sensitive data, and notify personnel to handle the threat. The

first step in implementing a successful DLP method is to identify sensitive data inside the company. Rules for DLP can be applied more successfully once the malicious insider threat is detected.

- Audit set-up:
 - Auditing is a key capacity that is frequently ignored or applied in a limited way, obstructing an organization's efforts to uncover hostile or irresponsible insider activities. However, once auditing is enabled, tools (such as a SIEM) must be in place to aid in the analysis, correlation, and alerting on events of interest. The information must also be protected against manipulation or erasure. If breaches or insider events may occur then it will be recorded. In an organization, it is oblivious to the malicious conduct, due to the large amounts of audit data without a method of analysis and warning reduces the value of the audit data.
- Privileged Access Management (PAM)
 - PAM systems provide a lot of preventative features and the detection features including session recording and auditing. This feature will keep track of all privileged user activity (such as given commands).

1.8 APPLICATIONS OF MITIGATING MALICIOUS INSIDER THREAT

The existing applications for mitigation of the malicious insider threat are explained below:

- Implement DLP
- Limit privileged access
- Implement User and Entity Behavior Analytics (UEBA)
- Segregate duties
- Block access to cloud storage sites
- Implement Multi-factor Authentication (MFA)
- Limit access to sensitive data
- Encrypt sensitive data
- Back up your data

These are some of the applications that are utilized for mitigating the malicious insider threat in an organization.

1.9 PROBLEM DEFINITION

- To develop a scientific method to detect and mitigate the Malicious Insider threats in cloud Environment.

1.10 OBJECTIVES

The major objectives are to

- Detect the Malicious Insider threat in cloud Environment using Machine Learning Technique
- Mitigate the Malicious Insider threat in cloud Environment using Multi Factor Authentication
- Enhance the security of the cloud

1.11 ORGANIZATION OF THIS REPORT

The rest of this report is organized as follows. In chapter 2, the related study for insider threat detection and mitigation framework is reviewed. In chapter 3, the novel malicious insider detection and mitigation framework is explained with the experimental results in detail. In chapter 4, the research is concluded with interesting future research ideas.

SUMMARY

The introduction about the malicious insider threat was elaborated in this chapter. In the upcoming chapter, the literature study for detection and mitigation of malicious insider threat is discussed.

LITERATURE REVIEW

CHAPTER 2

LITERATURE REVIEW

This chapter describes the previous works done in field of detecting and mitigation the malicious insider threat in cloud environment. The key concern is detecting and mitigating malicious insider threats using supervised machine learning-based anomaly detection technique and keystroke-based biometric authentication with OTP-based secondary authentication.

2.1 Study of Class Imbalance Problem

The major focus is to solve the class imbalance problem in imbalanced CERT data that comprise of malicious insider threat. The following table 2.1 describes the work done in the area of various sampling techniques.

Table 2.1: Study of Various Sampling Techniques

S.no	Author(s)	Sampling Techniques	Classification Algorithm(s)	Observations(s)
1.	Gosain and Sardana (2017)	SMOTE, ADASYN, Borderline-SMOTE, safe level-SMOTE.	Naïve bayes, SVM and Nearest Neighbor	Safe Level SMOTE performed better than other oversampling techniques based on f-measure and g-mean [2].
2.	Dittman et al. (2014)	RUS, ROS, SMOTE	KNN, SVM	RUS classified better than other techniques in SVM and KNN based on AUC-curve [3]
3.	Junsomboon and Phienthraku I (2017)	Neighbor Cleaning Rule (NCL), SMOTE	Naïve Bayes, Sequential Minimal Optimization (SMO) and KNN	Combined NCL and SMOTE provided a better result than SMOTE, NCL and ordinary data in various classifiers based on recall measures [4]

4.	Hasanin and Khoshgoftar (2018)	RUS	Random Forest	The minority class between 0.1% to 1% true positive rate is outperformed than 10% and 100% of class balanced data [5]
5.	Haibo He et al. (2008)	ADASYN and SMOTE	decision tree	The ADASYN algorithm provided better accuracy than SMOTE [6].
6.	Yap et al. (2014)	ROS, RUS, AdaBoost	Classification and Regression Tree (CART), C5 and Chi-Square Automatic Interaction Detection (CHAID)	RUS outperformed the other sampling techniques in three Decision Tree algorithm based on accuracy, sensitivity, specificity and precision [7].
7.	Fujiwara et al. (2020)	ADASYN, SMOTE, AdaBoost, RUSBoost, hyperSURF, HUSBoost and proposed HUSDOS-Boost sampling	Random Forest	HUSDOS-Boost outperformed the RUSBoost and provided 0.69% of G-mean to detect stomach cancer with the minority class instance less than 30 [8].
8.	Bunkhumpornpat and Subpaiboonkit (2013)	Improved SMOTE, Borderline-SMOTE and Safe-Level-SMOTE	Naive Bayes, Decision tree, KNN and RIPPER	Improved SMOTE provided a better result than other techniques on various classifiers and achieved 73% of F-measure and 78% of AUC [9].
9.	Abdi and Hashemi (2015)	Mahalanobis Distance-Based Over-Sampling Technique (MDO), SMOTE, Borderline-	Decision Tree, KNN and RIPPER	MDO performed better than other techniques with various classifiers in terms of MAUC and precision [10].

		SMOTE, and ADASYN		
10.	Elhassan and Aljurf (2016)	Tomek's Link(T-Link), RUS, ROS and SMOTE	SVM, ANN, Random Forest (RF) and Logistic Regression (LR)	T-Link performed best among various classifiers based on F-statistic, G-mean and AUC [11].

From the above table, it is observed that the different sampling techniques are applied to handle the class imbalance problem. Hence, to improve the correct detection of a malicious insider in an organization the different sampling techniques are implemented and compared.

2.2 Study of Malicious Insider Threat Detection

The work done in various field of malicious insider threat detection in cloud environment is summarized below:

Jiang et al. (2018) applied the user behaviour analysis using XGBoost and it is observed that it outperforms other algorithms namely SVM and Random Forest (RF) based on F-measure up to 99.96% to detect the malicious activity using CERT dataset. According to Kim et al. (2019), utilising Parzen window density and PCA to model user behaviour and find anomalies outperformed other techniques namely Gaussian density estimation to detect hostile insider threats. The novel behaviour analysis is proposed by (Liu et al, 2020) that uses Doc2vec to simplify the identification of insider threats using geographical and temporal metrics.

Eberle and Holder. (2009) implemented graph-based anomaly detection using the MDL algorithms namely GBAD-MDL, GBAD-P (probability) and GBAD-MPS (maximum partial substructure). It is observed that it identifies the graph-based anomalies such as email, phone traffic and business process to detect the insider threat than Probability and MPS algorithm [15].

Jiang et al. (2019) suggested the novel graph convolutional network algorithm for detection of malicious insider threat. In terms of accuracy, precision, and recall, it outperforms other algorithms such as random forest (RF), support vector machine (SVM), logistic regression (LR), and convolutional neural network (CNN) in detecting malicious insider and fraud activities.

In the integrated graph-based anomaly detection framework proposed by Gamachchi et al. (2017), isolation forest identifies 79 percent of individuals as genuine users and 31 percent as malicious insiders with questionable behaviour.

According to Le and Heywood. (2021), the unsupervised ensemble-based anomaly detection using autoencoder performs well than the other algorithms such as Isolation Forest, Lightweight on-line detector of anomalies (LODA) and Local Outlier Factor (LOF) based on voting metrics to detect the malicious insider threat. In machine learning techniques, bagging and boosting algorithms [24] also recommended to detect the malicious insider threat in cloud environment. Using data from US-CERT, Liu et al (2018) claim that the deep auto encoder detects all dangerous insider behaviour with a low false positive rate.

As a mitigation method, visual analytics [23] is advised for detecting malicious insider threat activities based on profile behaviour and specified attributes.

According to Diop et al (2019), ensemble learning behaviour based on the Gbc algorithm works well in detecting malicious insider activity. It outperforms other algorithms such as IForest, One-Class SVM, Local outlier factor (LOF), Elliptic envelope (EE), artificial neural network (ANN), Gaussian naive Bayes (Gnb), Bagging classifiers (Bgc), random forest (RF), and gradient boosting (Gbc) in both unsupervised and supervised learning-based testing, with accuracy ranges from (75 percent -99 percent). Following that, an ANN produced the accuracy range from (60% -99 %) outcomes in both tests.

Using numerous methods with computer log activity in a real organization database is another way to detect the hostile insider threat (Senator et al, 2013). This method suggests IP Thief Ambitious Leader Scenario Detector, File Events Indicator Anomaly Detection, Relational Pseudo Anomaly Detection, Repeated Impossible Discrimination Ensemble and Grid-based Fast Anomaly Discovery given Duplicates (GFADD). Table 2.2 describes the work done in the field of various supervised machine learning-based anomaly detection.

Table 2.2. Study of variants of machine learning-based anomaly detection

S.no	Author(s)	Algorithms applied	Observations
1.	Jiang et al. (2018)	XGBoost, SVM, Random Forest (RF)	User behaviour analysis using XGBoost outperforms other algorithms based on F-measure up to 99.96% to detect the malicious activity using CERT dataset [12]
2.	Eberle and Holder. (2009)	GBAD-MDL, GBAD-P (probability) and GBAD-MPS (maximum partial substructure)	Graph-based anomaly detection using the MDL algorithm identifies the graph-based anomalies such as email, phone traffic and business process to detect the insider threat than Probability and MPS algorithm [13]
3.	Liu and et. (2018)	Deep Autoencoder (AE)	Deep A.E. detects all malicious insider activity with a reasonable false positive rate using US-CERT data [14]
4.	Diop and et. (2019)	IForest, One-Class SVM, Local outlier factor (LOF), Elliptic envelope (EE), artificial neural network (ANN), Gaussian naive Bayes(Gnb), Bagging classifiers (Bgc), random forest (RF) and gradient boosting (Gbc)	Ensemble learning behavior using the Gbc algorithm outperforms other algorithms by (75%-99%) in both unsupervised learning-based testing and supervised learning-based testing. An ANN followed this with (60%-99%) results in both tests [15].
5.	Jiang et al. (2019)	RF, SVM, Logistic Regression (LR), Convolutional Neural Network (CNN), Graph Convolutional Network (GCN)	GCN performs better than other algorithms based on accuracy, precision and recall to detect malicious insider and fraud activities [16].

S.no	Author(s)	Algorithms applied	Observations
6.	Kim et al. (2019)	Gaussian density estimation, Parzen window density, Principal component	User behavior modelling and anomaly detection using Parzen and PCA provided a better result than other algorithms to detect malicious insider threats [17].
7.	Senator et al. (2013)	IP Thief Ambitious Leader Scenario Detector, File Events Indicator Anomaly Detection, Relational Pseudo Anomaly Detection, Repeated Impossible Discrimination Ensemble, Grid-based Fast Anomaly Discovery given Duplicates (GFADD)	The multiple methods detect the malicious insider threat using computer log activity in an actual corporate database [18].
8.	Lv et al. (2018)	Isolation Forest	MURB outperforms the ADAD with 80% precision and accuracy for detecting the malicious insider threat using CERT data [19].
9.	Gamachchi and et. (2017)	Isolation Forest	The combined graph-based anomaly detection framework identifies 79% of individuals as Genuine users and 31% as malicious insiders with suspicious activity [20].
10.	Liu et al. (2020)	Behaviour analysis	The new behaviour analysis framework named Doc2vec simplifies insider threat detection based on spatial and temporal metrics [21].
11.	Le and Heywood.	AutoEncoder, Isolation Forest, Lightweight on-	Unsupervised ensemble-based anomaly detection using Autoencoder outperforms the other algorithm

S.no	Author(s)	Algorithms applied	Observations
	(2021)	line detector of anomalies (LODA), Local Outlier Factor (LOF)	based on voting metrics to detect the malicious insider threat [22].
12.	Legg (2015)	Visual Analytics	Visual analytics is recommended to detect malicious insider threat activity based on profiling behaviour and selected features as a mitigation strategy [23].
13.	Singh and Ranga. (2021)	Boosted tree, bagged tree, subspace discriminant and RUSBoost	It achieves 97.24% accuracy in detecting the intruders in the cloud environment, and it outperforms the other existing techniques [24].

According to Sumitra et al. (2014), the key requirement for cloud security is a robust user authentication method that restricts unauthorised access [25]. It should be robust enough to defend the cloud against a variety of authentication attack, including malicious insider threats. In terms of selecting the strong authentication method, the multifactor authentication mechanism using biometric based keystroke dynamics and OTP based secondary authentication is applied.

2.3 Study of Biometric Authentication

The work done in various field of biometric authentication and multifactor authentication in cloud environment is summarized below:

Keystroke authentication is performed using CMU dataset in pertained CNN model with Resnet and Alexnet. Feature extraction using SVM with PCA, according to Tewari & Verma, (2022). Artificial Intelligence plays major role in biometric based user authentication using Hybrid Nanogenerators in Keystroke Dynamics (Maharjan et al., 2021). For cloud computing, the user authentication using dynamics keystroke service uses fixed-text written on a PC with a touch screen keyboard (Abo-alianet al., 2016).

According to Pukaret al. (2021), ANN outperforms SVM in terms of user identification and authentication in hybrid sensors-based keystroke authentication (TENG+EMG) using a shared password. Therefore, three classifiers namely Manhattan distance, Euclidean distance, and

Mahalanobis distance are applied for user keystroke authentication to achieve minimal false acceptance rate (Sae-Bae & Memon, 2022).

By exploiting piezoelectric keystroke dynamics for user authentication, RF surpasses conventional machine learning methods namely SVM and ANN to achieve EER of 72% (Huang et al., 2020). The security is accomplished by utilizing the piezoelectric force touch panel to enable keystroke dynamics and machine learning techniques to authenticate the user (Huang et al., 2020).

Based on the keystroke dynamics pattern, (paynath et al, 2018) correctly validates the user using proposed NeuroEvolution of the augmenting topology (P-NEAT) and obtained recognition rate 99%.

According to (Shi et al, 2020), the new user authentication mechanism in WIFI network is proposed that implements deep learning approach namely CNN. It tracks the daily user behavior of 11 participants in the wifi network and obtained 95% accuracy.

To reduce the malicious insider threat utilising keystroke authentication, the existing study exhibits superior outcomes by retraining to raise 65.91% the number of keys impostor typed before being discovered (Bondada & Bhanu., 2014). The authentication for static and continuous keystroke (chen et al., 2021) utilising the Gaussian model based anomaly detection is recommended for validating the user in online examination.

For biometric authentication utilising keystroke dynamics, enhanced security is obtained using mean and standard deviation. In comparison to previous algorithms, it lowered FAR and FRR (Jadhav et al, 2017). Using an ensemble of deep neural networks in keystroke dynamics, continuous authentication may be performed (Aversano et al., 2021). Deep neural networks and hierarchical multiple classifiers with random and k-means are used to achieve this.

Machine learning and deep learning are two alternative ways to authenticate the user in fixed text keystroke dynamics (Chang et al, 2021). Random forest, SVM, KNN, xgboost, LSTM, CNN, RNN, and MTP are some of the machine learning algorithms used for keystroke authentication. These algorithms also capable of determining the anonymous user's age and gender based on the user's normal pattern and mouse movement (Pentel., 2017). Krishnamoorthy

et al. (2018) used machine learning techniques to authenticate users using a virtual keyboard on a touch screen mobile phones. Thapliyal et al. (2022) used Samsung On7 Pro C3590 with 25 typical users to test their innovative user authentication system.

Using machine learning approaches, the combination of gait pattern and keyboard dynamics may be utilised to continually authenticate the user in smartphone authentication (Lamiche et al, 2019). However, the feature selection strategies are likewise effective for free-text keystroke dynamics authentication (Shanmugapriya & Padmavathi, 2011). UNAGI, a sensor-enhanced innovative biometric technique based on KNN Manhattan scaled weighted, has been proposed for user authentication in mobile phones (C. Giuffrida et al., 2014). A two-tier user authentication system that improves security and usability by leveraging keystrokes and paraphrase (Bhana & Flowerday, 2020).

Table 2.3 describes the work done in the field of various biometric authentication using keystroke dynamics.

Table 2.3. Study of various biometric authentication using keystroke dynamics

S.no	Author(s)	Algorithms applied	Observations
1.	Tewari & Verma. (2022)	Convolutional neural Network, SVM + PCA	It applies the CMU dataset and converts it into image format. Resnet and AlexNet are a pretrained CNN model. It is further classified using SVM + PCA for feature extraction. It is observed that ResNet outperforms and provides the best result in terms of 98% accuracy [26].
2.	(Maharjan et al., 2021)	ANN, SVM	In Keystroke Dynamics, Artificial Intelligence plays a key role in biometric-based user authentication utilising Hybrid Nanogenerators. The performance of ANN is better

			than SVM and achieves 99% accuracy for user identification and authentication using hybrid sensors(TENG+EMG)using a common password [27].
3.	Sae-Bae & Memon. (2022)	Manhattan distance, Euclidean distance, and Mahalanobis distance.	It is observed that less number of higher distinctiveness than lower distinctiveness and applied in all three classifiers based on distinctiveness score. This score indicates the FAR [28].
4.	Thapliyal et al. (2022)	k-nearest neighbors (k-NN) with fuzzy logic	The proposed novel user authentication mechanism was implemented in Samsung On7 Pro C3590 with 25 users. It performs well and achieved 1.88% EER [29].
5.	Huang et al. (2020)	Support vector machine, Artificial neural network and Random Forest	RF outperforms other machine learning algorithms and achieves an EER of 72% for user authentication by utilizing piezoelectric keystroke dynamics [30].
6.	Baynath et al. (2018)	Fuzzy Expert System (FESs), NeuroEvolution of the augmenting topology (NEAT), Proposed NEAT (P-NEAT), Support Vector Machine (SVM) and Chaotic Neural Network	P-NEAT outperforms other algorithms and achieves a 99% Recognition Rate (RR). It successfully verifies the user based on the keystroke dynamics pattern [31].
7.	Shi et al. (2021)	CNN	The CNN based user authentication system uses WIFI. It achieves authentication accuracy of 94%

			using 11 subjects based on daily activities [32].
8.	Bhana & Flowerday. (2020)	Shannon Entropy theory, Chunking theory and Keystroke level model	It proposed a two-tier user authentication solution, consequently reducing the false positive rate and potentially improving the security and usability of the user authentication process using keystrokes and paraphrasing [33].
9.	Abo-alianet al. (2016)	Outlier detection using Modified z-score, Fisher's linear discriminant, Quickly typed digraphs, Information gain, k-medoids clustering, SVM, naive Bayesian, multi-layer perceptron.	The proposed system results for dynamics keystroke user authentication service for cloud computing. It achieves 0.051 EER in free-text type, 34 ms verification time, and 53 ms verification time in fixed-text typed on the PC and touch screen keyboard [34].
10.	Bondada & Bhanu. (2014)	SVM	The proposed work shows better results by retraining to increase 65.91% the number of keys imposter typed before being caught to mitigate the malicious insider threat using keystroke authentication [35].
11.	Chenet al. (2021)	Gaussian model-based anomaly detector and keystroke stream processing, Keystroke Static Authentication and Keystroke continuous authentication.	The result obtained from the experiments using three public data sets and a case study for an online examination system achieves higher accuracy and efficiency. It achieves 0.05% FAR and 3.43% ERR to verify users using keystroke dynamics [36].

12.	Jadhav et al. (2017)	Mean and Standard Deviation	The proposed model provides advanced-level security and gives a FAR and FRR of 1% and 4%, respectively, for user authentication using keystroke dynamics [37].
13.	Aversano et al. (2021)	Deep neural network, hierarchical multiple classifiers with random and k-means	Using K-Means for user allocation and Bayesian voting for the final decision, it obtained the best performance. It achieved an overall accuracy of 99.7% for Continuous authentication using deep neural networks ensemble on keystroke dynamics [38].
14.	Chang et al. (2021)	XGBoost, multi-layer perceptrons (MLP), K-NN, SVM, CNN, RNN, LSTM	XGBoost outperforms and achieves the maximum accuracy at 96.39% for biometric authentication using fixed-text typing characteristics [39].
15.	Pentel. (2017)	Logistic Regression, SVM, KNN, C4.5, Random Forest	Random Forest performs better than other algorithms, but the result is still preliminary for classifying mouse and keystroke data from six different data sources to predict anonymous user age and gender [40].
16.	Krishnamoorthy et al. (2018)	SVM, Grid search optimization	SVM RBF outperforms SVM Linear to detect the user behaviour using biometrics authentication [41].
17.	Lamiche et al. (2019)	Multilayer perceptron (MLP), SVM, RF, Random Tree (RT), Naïve Bayes	MLP outperforms and achieves 99.11% accuracy with 0.684% false acceptance rate, 7% false rejection rate and 1% equal error rate. It applies a fusion method in feature

			level and sequential floating forward selection algorithm using a real-time dataset gathered from 20 subjects for smartphone authentication [42].
18.	Shanmugapriya, D & Padmavathi. (2011)	AntColony Optimization (ACO), Particle Swarm Optimization (PSO), Genetic Algorithm (GA), Extreme Learning Machine (ELM)	ACO with ELM outperforms other algorithms for feature subset selection to detect the optimum feature by reducing 46.51% of total features for free-text user authentication [43].
19.	Giuffrida et al. (2014).	Mean Euclidean, Mean Euclidean weighted, Mean Euclidean normed, Mean Euclidean normed weighted, Mean Manhattan, Mean Manhattan weighted, Mean Manhattan scaled, Mean Manhattan scaled weighted, KNN Euclidean, KNN Euclidean weighted, KNN Euclidean normed weighted, KNN Manhattan, KNN Manhattan weighted, KNN Manhattan scaled, and KNN Manhattan scaled weighted.	The sensor-enhanced novel biometric mechanism named UNAGI using KNN Manhattan scaled weighted performed well and outperformed other algorithms. It satisfies the EER of 0.08% [44].
20.	Huang et al. (2020)	SVM, ANN, RF	The dataset features, namely user touch and force, are extracted using a piezoelectric force touch panel for user authentication. It attained an Equal Error Rate (EER) of 0.720% using RF, which outperforms other

			applied ML techniques [45].
--	--	--	-----------------------------

The class imbalance can be solved using different sampling techniques as mentioned in table 2.1. From the table 2.2, the different types of machine learning algorithms are explored to detect the malicious insider threat. Table 2.3 shows that keystroke based biometric authentication applies different machine learning algorithms to verify the user in login authentication.

2.4 Observations from literature

From the literature study, the following observations are encountered to propose a framework for detection and mitigation of mi.

1. Previous works has addressed the undersampling and oversampling technique to handle the class imbalance problem for better classification.
2. The machine learning techniques are limited to insider threat detection in cloud environment. So, the supervised machine learning based anomaly detection techniques can be explored.
3. The combination of multi-factor authentication for mitigation of malicious insider threat is not addressed in previous work. Thus, to mitigate the malicious insider threat, keystroke based biometric authentication and OTP based secondary authentication can be applied in cloud environment.

SUMMARY

In this chapter, the study of class imbalance problem, malicious insider threat detection and keystroke based biometric authentication were elaborated. In the next chapter, the proposed methodology for detection and mitigation of malicious insider threat in cloud environment is explained.

METHODOLOGY

CHAPTER 3

METHODOLOGY

This chapter elaborates the proposed methodology for detection and mitigation of malicious insider threat in cloud environment. The proposed methodology for malicious insider detection and mitigation is shown in figure 3.1.

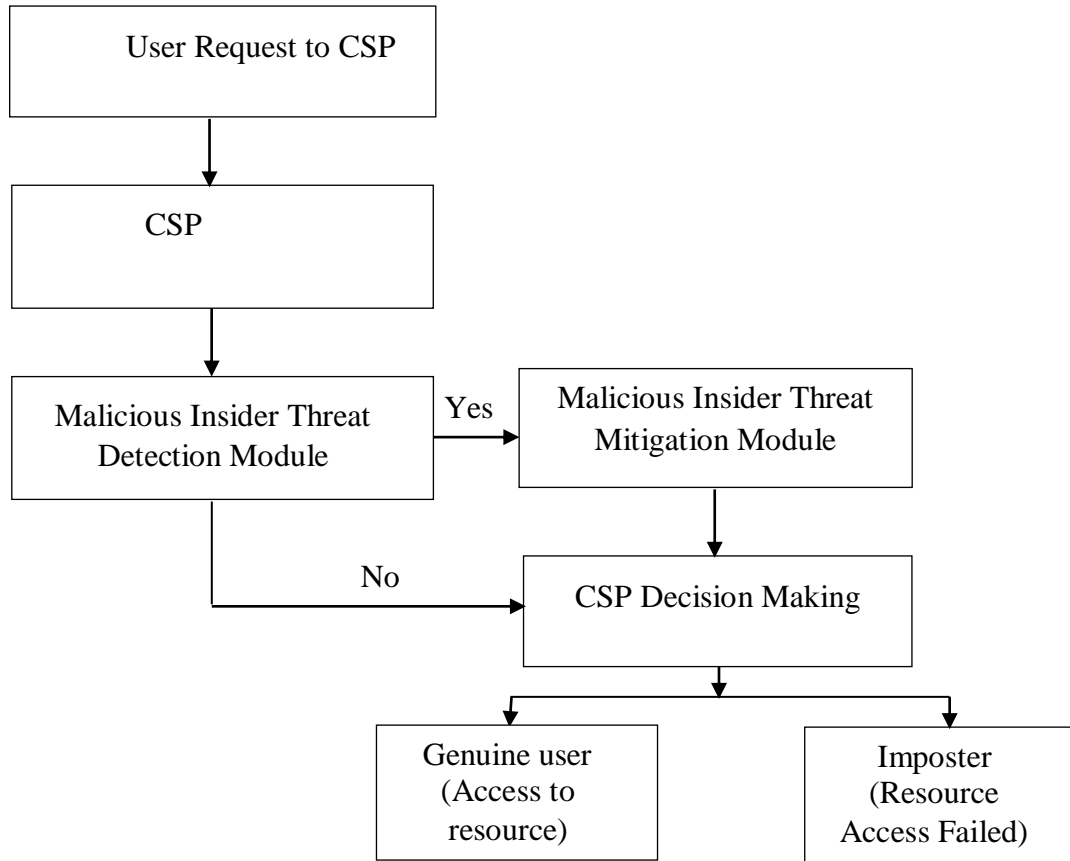


Figure 3.1. Proposed Framework

The methodology has two phases. i.e., malicious insider threat detection phase and malicious insider threat mitigation phase. It implements anomaly detection using machine learning such as One-Class Support Vector Machine (OCSVM) for detecting the malicious insider and Multifactor Authentication using biometric authentication and secondary authentication for mitigating the malicious insider.

3.1 Phase I: DETECTION PHASE

Figure 3.2 shows the overview of malicious insider threat detection phase. It is sub-divided into three phases. Sub-phase I demonstrate the dataset used. The pre-processing techniques to handle the imbalanced class problem are explained in sub-phase II. In sub-phase III, supervised learning-based anomaly detection namely OCSVM is applied to detect the malicious insider threat in the detection phase.

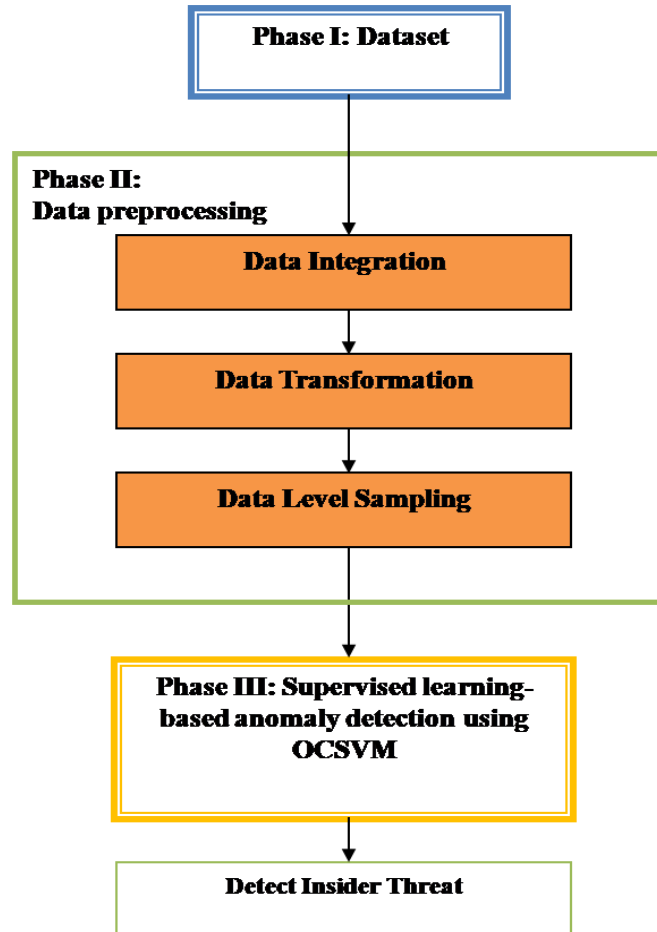


Figure 3.2: Overview of malicious insider threat detection phase

3.1.1 Sub-Phase I: Dataset

The "CERT Insider Threat Tools" dataset (Carnegie Mellon's Software Engineering Institute, Pittsburgh, PA, USA) [1] is used to perform the framework as mentioned above. The CERT dataset is an artificially generated synthetic dataset to validate insider-threat detection frameworks [36]. They are (i) employee activity log in the computer such as logon, device, http,

file and email. The activity information comprises ID, timestamps, user ID, pc ID and activities.

(ii) Organization structure information such as employee departments and roles. The US-CERT dataset has six major versions (R1 to R6) [47]. Version R3 has two distinctions. They are R3.1 and R3.2. It includes the information on activity, employees and malicious insider activities based on dataset variation. In this research, version R3.1 is considered as a baseline. It includes 1000 users, among two employees performed maliciously. The log information that satisfies the selected scenarios are http, logon and device. It contains a unique id, date, user id, pc id, and activity. The information includes employee name, user id, email, role, project, business unit, functional unit, department, team, and supervisor in the LDAP folder. The dataset version r3.1 is considered as a primary dataset to analyse and detect the malicious insider threat. Some malicious insider threat-based scenarios [24] are defined below:

- Scenario 1: An individual in an organization working after working hours, often used to carry a removable drive and uploaded the important information to wikileaks.org. Later resigned from the organization.
- Scenario 2: An individual in an organization visited job websites and beseeched employment opportunities from a competitor of the business. The abnormal behaviour of the employee increases in data transfer using removable drives. Later resigned from the organization.
- Scenario 3: Unauthenticated or unsatisfactory system administrator tries to install malicious software to collect sensitive information and utilize the removable drives for data transmission from the particular authorized system. Gather sensitive information to access the authenticated system. It also contains emails regarding sensitive information in an unusual manner in an organization. Later, resigned from the organization.
- Scenario 4: Over three months, individuals frequently logged into other user's computers. Searched and forwarded files to a personal email address.
- Scenario 5: Uploaded documents to Dropbox for personal gain.

This project considers scenario-1 and scenario-2 among the above mentioned five scenarios. So, the primary data related to selected scenarios are considered as Base data and others are neglected.

3.1.2 Sub-Phase II: Data pre-processing

The base data undergoes three pre-processing steps to make the data suitable for classification. It includes data integration, data transformation and data sampling.

a) Data Integration:

Detection of Malicious insider threat records related to a device status, login status and HTTP operation that satisfies the above-selected scenario. In the selected dataset, the activities of employees are warehoused in three tables such as logon, device and HTTP. The different tables must be combined as a single homogeneous employee activity table. The selected records are integrated using simple feature concatenation techniques. While other records are neglected. The following table 3.1 demonstrates the feature details of integrated data. The following table 3.1 demonstrates the feature details of integrated data.

TABLE 3.1. FEATURE DETAILS OF INTEGRATED DATA

Features	Description
InsiderThreat	It considers malicious activity or not
Vector	It is considered the origin of data
date	Date of the particular event
User	User id who carries particular activity
Pc	Unique identification for each computer
Activity	The action of a particular user

b) Data Transformation

The integrated data needs to be transformed to a categorical value for further processing. The features namely 'vector', 'pc', 'user' and 'activity' from integrated data converted into a numerical value. The value of 'date' is converted into the number of epochs. The following table 3.2 shows the details of transformed data.

Table 1.2.Transformed data

Features	Before Transformation	After Transformation
InsiderThreat	1	1
Vector	Logon	0
date	07-01-2010 02:23:00	1280707200
User	CCH0959	4
Pc	PC-0588	128
Activity	http://linkedin.com/jobs/displayhome.html	750

c) Data Level Sampling

Since the transformed data have a massive number of non-malicious class instances than the instance of a malicious class, it encounters the imbalanced class problem. The training of imbalance class instances will cause an insignificant result in detecting the malicious insider threat in an organization. It is necessary to balance the instance in all classes for accurate classification. To solve the class imbalance problem, three different types of techniques have been used. They are Data level solution, Algorithmic level solution, Ensemble-based learning solution [25]. The solution at the data level for the class imbalance problem is based on sampling methods. This technique provides the solution by altering the pattern of data distribution. It is also said to be restructuring the class imbalanced data to make it well-balanced data. It is accomplished by both undersampling and oversampling.

The different types of oversampling techniques are Synthetic Minority Over-Sampling Technique (SMOTE), Adaptive Synthetic (ADASYN) and Random Oversampling (ROS). Some of the essential undersampling techniques are Edited Nearest Neighbours (ENN), Near-Miss 1 (NM-1), Near-Miss 2 (NM-2), Random Under sampler (RUS), Tomek-link (T-L) have been implemented.

In the pre-processed dataset, a feature like 'InsiderThreat' is the target variable where majority class instance '0' denotes non-malicious activity and minority class instance '1' denotes malicious activity. It is difficult to classify the minority class because the minority class instance is lesser than the majority class instance. So, it arises a class imbalance problem during classification where the data is distributed unequally for all the classes. This results in misclassification and misinterpretation of data. To handle the class imbalance problem, the data level solution such as oversampling and undersampling techniques are recommended.

I) Oversampling techniques

The major focus of Oversampling technique is to replicate the instance of the minority class until the dataset is balanced. Since the size of minority class instances would increase abruptly, the learning time also increases. This research considers the three oversampling algorithms to resample the imbalanced data. They are ROS, SMOTE and ADASYN. One of the standard techniques in oversampling is ROS. It multiplies the instance of the minority class randomly by replicating the minority class instance. Thus, it raises the problem known as overfitting. To overcome the overfitting [3] [7] [11], artificial synthetic methods are recommended.

In SMOTE eqn. (1), a new artificial synthetic dataset is generated by combining minority class instance x_i and interpolation within KNN, namely x_{zi} [2] [3] [6] [8] [10] [11]

$$x_{new} = x_i + \lambda(x_{zi} - x_i) \quad (1)$$

Where the λ is denoted as a random number between 0 and 1, the balanced data is created by interpolation between x_i and x_{zi} . Minority instances are generated using (i) Regular. (ii) Borderline approach using KNN (iii) SVM approach [9]. SMOTE modifies the artificial instance of minority class based on weight for each class is called ADASYN. It generates several instances for minority classes proportional to the number of the adjacent class instance [2] [6] [7] [8] [9]. It concentrates on outlier or minority class instances.

II) Undersampling technique

The primary focus of the Undersampling technique is to eliminate the instance of the majority class until the dataset is balanced. The decrease in the size of the majority class instance decreases the learning time [5]. This paper focuses on five undersampling algorithms used to balance the class imbalanced data. They are RUS, NM-1, NM-2, T-L and ENN.

One of the standard techniques in undersampling is ROS. It minimizes the instance of the majority class in a random pattern until the majority class instance equals the minority class instance [5] [7] [9] [11]. Hence it causes a loss of important information in the majority class. Near-miss's idea is to resample the instance of the majority class necessary to differentiate all classes. In NM-1, the majority class instance is selected if it satisfies the minimum average distance for N neighbouring minority class instance. In NM-2, the majority class instance is selected if it satisfies the minimum average distance for N outermost minority class instance. T-L's objective [11] is to clean the majority instance by eliminating the outlier, the same as a classifier.

$$d(x,z) < d(x,y) \text{ or } d(y,z) < d(x,y) \quad (2)$$

Where d is defined as the distance between two instances, the link exists if the two instances of dissimilar classes are nearby. In ENN [11], the KNN eliminates the instance that fails to satisfy the neighbor. Table 3.3 illustrates the various sampling techniques and their working criteria.

TABLE 3.3: VARIOUS SAMPLING TECHNIQUES AND THEIR WORKING CRITERIA

Sno	Sampling technique	Working criteria
1	ROS	Multiplies the instance of the minority class randomly. Thus, it raises the overfitting problem
2.	SMOTE	A balanced new artificial synthetic dataset is generated by combining minority class instance x_i and interpolation within KNN, namely $x_{zi}x_{new} = x_i + \lambda(x_{zi} - x_i)$
3.	ADASYN	It generates several instances for minority classes proportional to the number of the adjacent class instance
4	RUS	It minimizes the instance of the majority class in a random pattern until the majority class instance equals the minority class instance
5	NM-1	The majority class instance is selected if it satisfies the

		minimum average distance for N neighboring minority class instance
6	NM-2	The majority class instance is chosen if it meets the minimum average distance for N outermost minority class instance
7	T-L	Eliminates the outlier link to reduce the majority instance.
8	ENN	The KNN eliminates the instance that fails to satisfy the neighbor.

3.1.3 Sub-Phase III: Supervised Learning-Based Anomaly Detection

To find the outlier in the data in numerous circumstances is essential for anomaly detection. Many machine learning models include errors when it comes to modelling outlier components. As a result, determining whether the new observation belongs to the same current distribution or should be classified as distinct becomes a key necessity. These detections are frequently employed in the cleaning of datasets. The two most critical responsibilities are outlined below.

Outliers are observations that differ significantly from the rest of the training data. Outlier estimators attempt to fit the majority of the training under an area while disregarding the deviating data. Unsupervised anomaly detection is another name for it. Support vector machines are a type of machine learning model that may be used for classification and regression analysis. It's mostly used to solve categorization challenges. Consider OCSVM modelling using data that is divided into two classes.

One of the most essential features of OCSVM is that it uses its nonlinear function to extend data with greater dimensions in space, resulting in nonlinear decision limits. It employs its function to raise the feature space F of observations in the I space that cannot be separated using a linear function or straight line. The straight hyperplane can split raised feature space. This hyperplane is used to divide data from one class from data from another. A nonlinear curve can be used to represent this hyperplane.

Slack variables are used to prevent the model from overfitting by allowing some data points to fall within the margin. And the constant C , which is always bigger than zero, specifies the trade-

off between maximising the margin and the quantity of training data points included inside it (and thus training errors).

Any hyperplane may be expressed as a set of X satisfying points in eqn. 3.

$$w^T x - b = 0, \quad (3)$$

Where w is a normal vector to the hyperplane (not necessarily normalised). The hyperplane shown above is a linear SVM in eqn. 4, in which the decision function for a data point x may be represented as

$$f(x) = \text{sgn}(\sum_{i=1}^n \alpha_i y_i K(x, x_i) + b) \quad (4)$$

The function $K(x, x_i)$ is a kernel function with the formula $k(x, x_i) = \phi(x)^T \phi(x_i)$.

In this case, is a nonlinear function. The dot vector of the vectors in the feature space determines the decision function's conclusion. This kernel function is a simplified version of any kernel function that may be used with data that has a basic spatial distribution. The gaussian radial basis kernel function is the most often used kernel function, and it is defined as:

$$K(x, x') = \exp\left(-\frac{\|x-x'\|^2}{2\sigma^2}\right) \quad (5)$$

Where the numerator is the dissimilarity function and the denominator is a kernel parameter in eqn. 5. We can classify data having two classes using this combination of methods, but to classify data containing only one class next.

In One-Class SVM, there are two alternative techniques. The first approach, proposed by Schölkopf et al, detects novelty by separating data points from the feature space and maximising the distance from the hyperplane to the feature space. The second approach, proposed by Tax and Duin, detects novelty by separating data points from the feature space and maximising the distance from the hyperplane to the feature space. This method produces functions that focus on the space where the density is highest, allowing the function to return +1 if the observation is in a dense zone and -1 if it is in a low dense region.

The minimization function for linear SVM models in eqn. 6 may be written as:

$$\min_{w,b,\varepsilon_i} \frac{\|w\|^2}{2} + C \sum_{i=1}^n \varepsilon_i \quad (6)$$

Subject to:

$$y_i(w^T \varphi(x_i) + b) \geq 1 - \varepsilon_i \text{ for all } i = 1, \dots$$

$$\varepsilon_i \geq 0 \text{ for all } i = 1, \dots$$

The minimization function is used in the eqn. 7.

$$\min_{w,\varepsilon_i,\rho} \frac{1}{2} \|w\|^2 + \frac{1}{vn} \sum_{i=1}^n \varepsilon_i - \rho \quad (7)$$

Subject to:

$$y_i(w \cdot \varphi(x_i)) \geq \rho - \varepsilon_i \text{ for all } i = 1, \dots$$

$$\varepsilon_i \geq 0 \text{ for all } i = 1, \dots$$

The parameters of these functions change somewhat in linear SVM for multiclass classification using the C parameter and this technique for one class using the parameters v. This parameter is used to establish the upper and lower bounds on the outliers % and the number of training instances that may be utilised for SVM modelling.

Where others employ the planner technique, the second approach uses a spherical boundary in the feature space. The radius and the centre make up the resultant hypersphere. Where the radius's square is the smallest. The hypersphere's centre is made up of a series of linear support vector combinations. This approach's minimization function is explained in eqn.8.

$$\min_{R,a} R^2 + C \sum_{i=1}^n \varepsilon_i \quad (8)$$

Subject to:

$$\|x_i - a\|^2 \leq R^2 + \varepsilon_i \text{ for all } i = 1, \dots$$

$$\varepsilon_i \geq 0 \text{ for all } i = 1, \dots$$

This strategy makes use of similarity. The difference of the function is the distance between the hypersphere's centre and data points that is strictly less than or equal to R , resulting in a dense space of data points, with distances above this deemed outliers. Slack variables are used with the parameter C to generate a soft margin.

The first approach uses parameters to distinguish between two-class and one-class SVM classifiers, while the second way uses the hypersphere to place one-class data beneath the sphere if the data point's distance from the hypersphere's centre is less than or equal to the radius.

In this project, the first approach for one class SVM for anomaly detection is used. The sampled data is well balanced to perform the insider threat anomaly detection. In the insider detection phase, OCSVM is used to train the balanced data. It is accomplished using supervised learning-based anomaly detection using OCSVM. The OCSVM classifier input the sampled data to predict the malicious scores for each activity. In this paper, OCSVM is used to predict the anomaly score based on the employee's activity. The employee with the highest anomaly score is considered a malicious insider.

3.2 RESULTS AND DISCUSSION IN PHASE I

3.2.1 Performance metrics

To evaluate the proposed methodology, Accuracy, Recall, Precision, F-score, True Detection Rate (TDR) and False Detection Rate (FDR) are used as performance metrics based on True positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN). Table 3.4 elaborates the performance metrics used in phase I.

Table 3.4: Performance metrics in phase I

Sno.	Performance metrics applied	Definition	Formula
1.	Accuracy	It is the most frequently used evaluation metrics and is defined as the proportion of accurately predicted instances to the overall sum of instances. It is derived in eqn. 3	$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} \quad (3)$
2.	Recall or True Detection Rate (TDR)	It is defined as a correctly predicted positive instance. It is also known as a measure of correctness. It is derived in eqn. 4	$\text{Sensitivity or Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (4)$

3.	Precision or False Detection Rate (FDR)	It is a measure of exactness to predict the positive instance as positive by eliminating the inaccurately predicted negative instance as a positive. It is derived in eqn. 5	Precision = $\frac{TP}{TP+FP}$ (5)
4.	F-score	It is the weighted average of precision and recall in binary classification. The increase in precision and recall increases the value of σ . It is derived in eqn. 6	F-score = $\frac{(1+\sigma^2) * Precision * Recall}{\sigma^2 * Precision + Recall}$ (6)

3.2.2 Experimental Analysis

The experimental analysis of data sampling and anomaly detection in phase I is discussed in this section.

i) Data level sampling

The performance of the SVM classifier is compared using above mentioned evaluation metrics. The following table 3.5 illustrates the performance of the SVM classifier after applying different oversampling and undersampling techniques. From table 3.4 it is observed that the performance of ADASYN, ROS and SMOTE remains the same and recall of a non-malicious event is less. Hence, the handling of imbalanced data using the oversampling technique is difficult. In the

undersampling technique, the f-score of NM-1 is almost less and specificity is unsatisfactory where it fails to detect the malicious activity. ENN and T-L achieved equal modest performance. NM-2 surpasses the ROS and improves the generation of an artificial minority class instance dramatically. Thus, it achieves high performance in Recall, Precision, F-score and Accuracy.

TABLE 3.5: PERFORMANCE METRICS OF EIGHT SAMPLING METHODS

	Accuracy	F score	Precision	Recall
Oversampling techniques				
ADASYN	0.680375	0.80± 0.03	0.99±0.02	0.67±0.77
ROS	0.680375	0.80±0.03	0.99±0.02	0.67±0.77
SMOTE	0.680375	0.80±0.03	0.99±0.02	0.67±0.77
Undersampling techniques				
ENN	0.680375	0.80±0.03	0.99±0.02	0.67±0.77
NM-1	0.319625	0.48±0.00	0.97±0.00	0.32±0.22
NM-2	0.84325	0.91±0.02	0.99±0.01	0.84±0.28

RUS	0.716625	0.83±0.04	0.99±0.02	0.71±0.74
T-L	0.680375	0.80±0.03	0.99±0.02	0.67±0.77

Table 3.6 demonstrates the comparison between the performance of the SVM classifier using imbalanced and balanced data. From table 5, it is observed that imbalanced data fails to detect malicious activity. The Recall of SVM classifier using balanced data correctly predicts the non-malicious activity than imbalanced data. Precision and f-score are outperformed using balanced data while accuracy remains satisfactory in the detection of malicious activity.

TABLE 3.6: COMPARISON OF SVM CLASSIFIER PERFORMANCE USING IMBALANCED AND BALANCED DATA

	Accuracy	F-score	Precision	Recall
Imbalanced data	0.991625	0.99±0.00	0.99±0.00	1.00±0.00
Balanced data	0.84325	0.91±0.02	0.99±0.01	0.84±0.28

The performance of undersampling techniques outperformed the oversampling techniques to handle the imbalanced CERT dataset using SVM Classifier. NM-2 works better than other sampling techniques based on F-score and recall. It eliminates majority class instances safely, resulting in improved performance than RUS, NM-1, ENN and T-L. The balanced data using NM-2 is used for further classification.

ii) Supervised Learning-Based Anomaly Detection

The novelty method applies the CERT dataset, as mentioned previously, to perform data pre-processing and anomaly detection using the OCSVM classifier. The implementation is done in a jupyter notebook using python language. Table 3.7 explains the simulation parameters of OCSVM.

TABLE 3.7: SIMULATION PARAMETERS OF OCSVM

Parameters	Values
Kernel	RBF
Gamma	0.001
Nu	0.02

The simulation process of the proposed novel methodology is as previously explained, and the result is explained below.

Figure 3.3 shows the sampled data consisting of genuine user and malicious activity in an organization. However, the genuine user's activity is lesser than the activity of the malicious user.

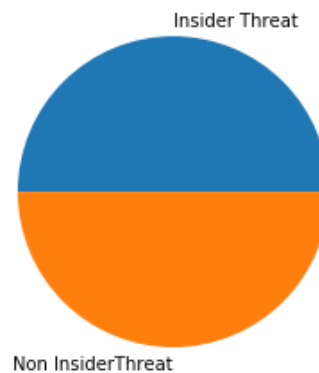


Figure 3.3: Balanced data of Genuine users and malicious user

Figure 3.4 illustrates that the total activity of the malicious user is seven times more than the activity of the genuine user. It is required to detect the malicious insider in an organization. The supervised machine learning algorithm such as OCSVM is implemented to train, classify and predict the malicious activity in an organization. The predicted value comprises 1 and -1, where -1 indicates the malicious activity. Based on the activity of each employee in an organization, the anomalous score is calculated. By altering the threshold value, the anomalies are filtered.

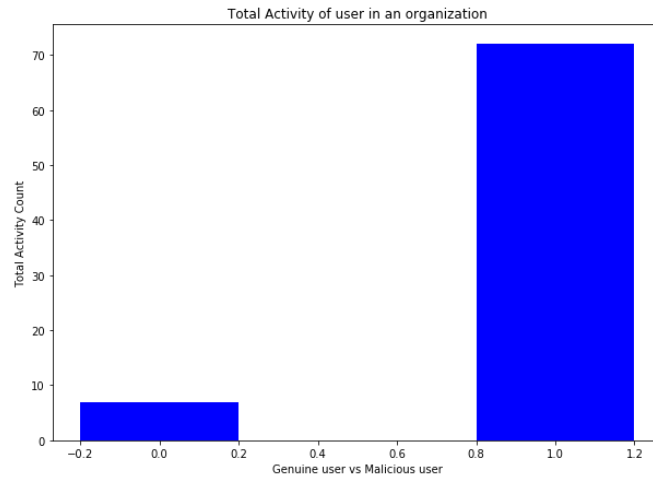


Figure 3.4: Activity count of genuine user and malicious user

Table 3.8 shows the threshold value, total filtered anomalous activity, true detection rate and false detection rate.

TABLE 3.8: DESCRIPTION OF TOP X% OF MALICIOUS ACTIVITY

Top X%	Threshold value	Total filtered activity (out of 268)	True detection rate(in term of %)	False detection rate(in term of %)
10%	0.0195312524	54	20.14%	0%
20%	0.0195362745	106	39.55%	0%
30%	0.0200169149	161	60.07%	0%
40%	0.0202734374	214	79.05%	0%
50%	0.0202734375	268	100%	0%
60%	0.0202734404	321	100%	19.77%
70%	0.0203840269	369	100%	37.68%
80%	0.0204303234	405	100%	51.11%
90%	0.0204303234	405	100%	51.11%

99%	0.0205078125	526	100%	96.26%
-----	--------------	-----	------	--------

True detection is more inconsequential in the top 10% and 20% of malicious activity. It detects 20% and 39% of malicious activity in an organization which is insignificant in detecting the crucial malicious insider. It detects 60% to 79% of the malicious activity of employees in the top 30% and 40% activity. In the top 50%, all the malicious activity is detected and achieves a 100% true detection rate. Since it does not detect the activity of the genuine user as the malicious activity, it achieves a 0% false detection rate. From the top 60% to the top 99%, the false detection rate increases abruptly while the true detection rate remains the same. Hence, to detect all the malicious activity in an organization with a 0% false detection rate, the top 50% with a threshold value of 0.202734375 is implemented.

Table 3.9 shows the activity of malicious employees in the top 50%. The employee, namely CCH0959, performed a total number of 242 malicious activities in an organization. At the same time, the employee CSF0929 performed 26 malicious activities in an organization.

TABLE 3.9: THE ACTIVITY OF MALICIOUS EMPLOYEES IN TOP 50%

Top 50%	User	Total number of malicious activities
	CCH0959	242
	CSF0929	26

Figure 3.5 demonstrates the detection of malicious insiders based on employee activity in an organization.

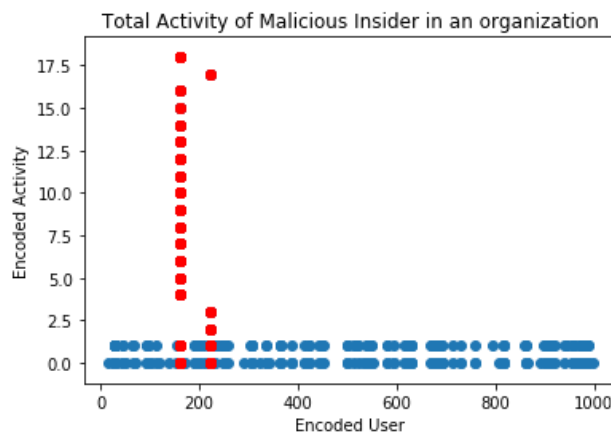


Figure 3.5: Malicious insider based on employee activity

Figure 3.6 shows that the employee labelled as a malicious insider is correctly detected in an organization. Figure 3.7 depicts the anomalous activity of malicious insiders detected. It shows that all the malicious activity of insiders has been detected without any false detection of genuine users.

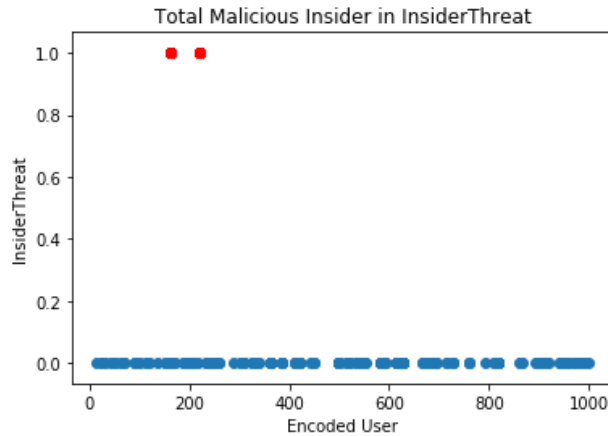


Figure 3.6: Malicious insider detected in an organization

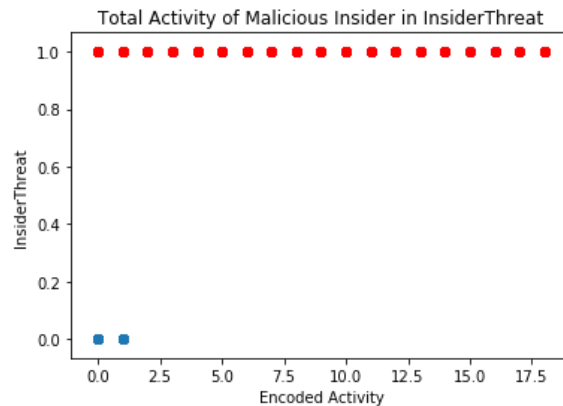


Figure 3.7: The malicious activity of malicious employees in an organization

The proposed insider detection model generated a 20.14% true detection rate in the top 10% of malicious activity with zero detection rate. In the top 30% and 40% of malicious activity, the true detection rate raised abruptly to 60.07% and 79.05% with no false detection rate. In the top 50%, the proposed framework achieved a 100% true detection rate with a 0% false detection rate and satisfies the performance to detect the malicious insider threat in the cloud environment.

3.3 Phase II: MITIGATION PHASE

After detecting the malicious insider threat, the next step is to mitigate the malicious insider threat in cloud. Figure 3.8 shows the overview of malicious insider threat mitigation phase. It is subdivided into two phases. Sub-phase I authenticate the malicious insider using keystroke based biometric authentication. In sub-phase II, the secondary authentication is carried out using OTP verification. The successful user who satisfies both the authentication is considered a genuine user and login into the system securely. Otherwise, the user is considered malicious and fails to log in.

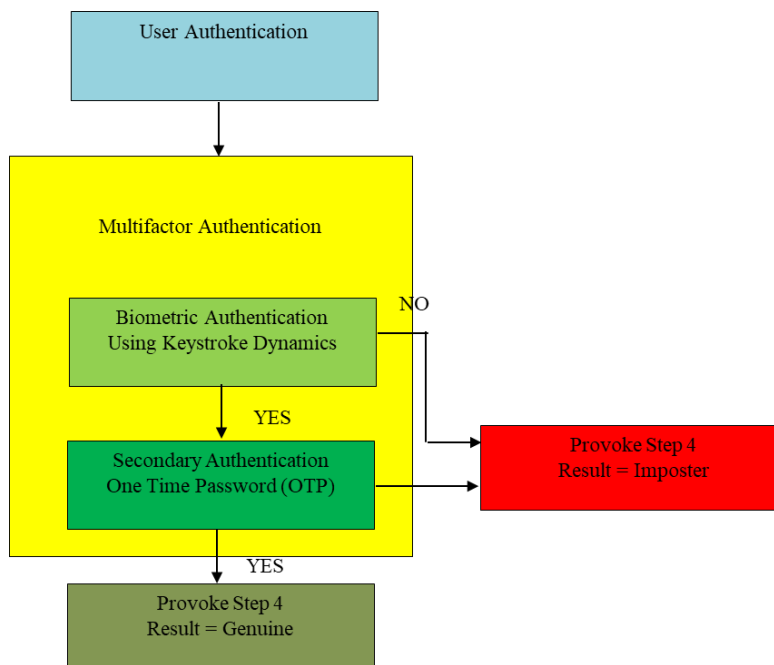


Figure 3.8: Overview of malicious insider threat mitigation phase

3.3.1 Sub-Phase I: Data Acquisition

The proposed mechanism is experimented using CMU Keystroke Dataset [46]. This dataset is a Public Typing Database Created and collected by Kevin Killourhy and Roy Maxion from Carnegie Mellon University. This dataset represents the typing pattern of 51 individuals typing the shared password, namely, ".tie5roanl". Each user types this password 50 times per session from the eight sessions. Dwell time, flight time and di-graph of every user were calculated. This obtained CMU dataset is split into train and test data.

3.3.2 Sub-Phase II: Biometric Authentication

Figure 3.9 shows the biometric authentication using keystroke dynamics, and it applies the Gaussian mixture model as a distance measure to differentiate the train and test data of the malicious insider.

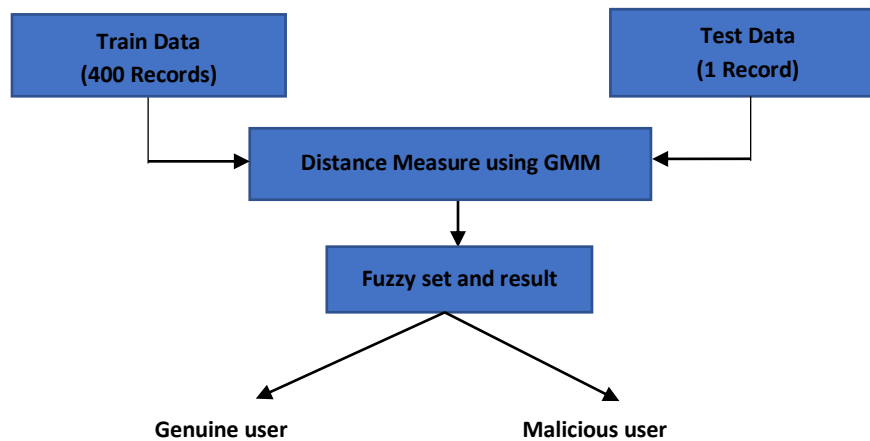


Figure 3.9: Overview of biometric authentication using keystroke dynamics

From the above figure, it is understandable that the keystroke behavior of a particular user is considered train data. The model is trained using Gaussian Mixture Model (GMM). The model is tested by selecting a random keystroke behavior (i.e. test data) to predict the user behavior. The biometric authentication model is measured using evaluation measures such as false acceptance rate (FAR), false rejection rate (FRR) and Equal error rate (EER).

Decision making using fuzzy logic

Fuzzy logic is exploited in artificial intelligence (AI) systems to simulate human thinking and cognition. Fuzzy logic allows 0 and 1 as extreme cases of truth, as well as different degrees of truth in between.

In the 1960s, Lotfi Zadeh of the University of California in Berkeley proposed the concept of fuzzy logic. Zadeh was working on a computer programme that could grasp natural language. Natural language, like most other aspects of life and the cosmos, is difficult to convert into absolute terms such as 0 and 1. Whether everything can be described in binary terms in the end is a philosophical subject worth exploring, but in fact, most of the data in a computer is somewhere

in between 0 and 1, as are the consequences of computing. It may be helpful to think of fuzzy logic as the true nature of thinking, and binary, or Boolean, logic as a subset of it.

A conditional statement within the fuzzy logic is a fuzzy rule. IF THEN statements define the shape of fuzzy rules. If y is B, then x is A, where x and y are linguistic variables and A and B are fuzzy set linguistic values. In fuzzy logic, these are the rules of inference that determine the value of an output variable depending on the values of input variables. Based on fuzzy logic, a logical rule is formed. IF-THEN language formulations of the general form "IF A THEN B," where A and B are (groups of) propositions with linguistic variables.

The fuzzy logic uses the three fuzzy rules using EER and two threshold values, i.e., Threshold1=0.35 and Threshold2=0.49. The fuzzy rules are defined below:

F1: $EER < \text{Threshold1} \text{ or } EER = \text{Threshold1} \rightarrow (1)$

F2: $\text{Threshold1} < EER < \text{Threshold2} \rightarrow (0)$

F3: $\text{Threshold2} < EER \rightarrow (-1)$

Where 1 is a genuine user, 0 is a Genuine like malicious user, and -1 is the malicious user. Based on the fuzzy rule (i.e. F1, F2 and F3), the output is segregated into malicious users, genuine like malicious and genuine users. The malicious user fails to perform secondary authentication, while others are considered for secondary authentication.

3.3.3 Sub-Phase III: Secondary Authentication

Two-factor authentication (2FA), often known as two-step verification or dual-factor authentication, is a security method in which users validate their identity using two independent authentication factors. 2FA is used to safeguard a user's credentials as well as the resources that is used for right to access. Single-factor authentication (SFA), in which the user gives only one factor, often a password or passcode, provides a better level of security than two-factor authentication (TFA). Two-factor authentication requires a user to provide a password as the first

factor and a second, distinct element, which is commonly a security token or a biometric factor, such as a fingerprint or face scan.

Because even if a user's password is compromised, a password alone is not enough to pass the authentication check, two-factor authentication act as an extra layer of protection to the authentication process, making it more difficult for attackers to obtain access to a person's devices or online accounts.

Figure 3.10 shows the secondary authentication using OTP to double verify the identity of genuine and genuine like malicious users as either genuine or malicious.

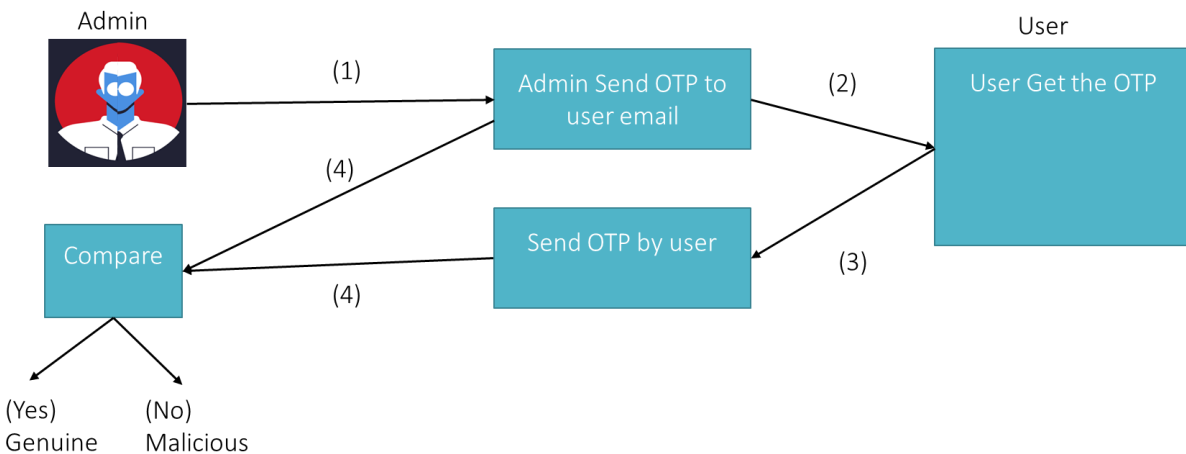


Figure 3.10: Secondary Authentication using OTP

The following steps are applied to verify the genuine user using OTP-based secondary authentication from the above figure.

Step 1: Randomly generate the OTP in secondary authentication protocol and send it to the email id of the genuine user.

Step 2: user receives the email with OTP for verification

Step 3: user type the received OTP in the secondary authentication protocol webpage for further verification

Step 4: The received OTP is compared with the actual OTP in the secondary authentication protocol webpage. If both the OTP are the same, then the authentication is successful. Otherwise, the authentication is a failure.

Thus, the user's identity is successfully verified as genuine and further, and the user can use login successfully into the cloud account. Thus it enhances cloud security and mitigates the malicious insider threat in the cloud environment.

Decision making using fuzzy logic

The fuzzy logic uses the three fuzzy rules using the status of both authentication, i.e., Authentication1 and Authentication2. The fuzzy rules are defined below:

F1: Authentication1 is “Genuine”, and Authentication2 is “Success” \rightarrow (1)

F2: Authentication1 is “Genuine”, and Authentication2 is not “Success” \rightarrow (0)

F3: Authentication1 is not “Genuine” \rightarrow (0)

Where 1 is a genuine user and 0 is malicious user. Based on the fuzzy rule (i.e. F1, F2, F3 and F4), the output of the proposed model is segregated into malicious users and genuine users. The genuine user can log in to the cloud security system successfully, whereas the malicious user fails the successful login and prompts to change the user password.

3.4 RESULTS AND DISCUSSION IN PHASE II

3.4.1 Performance metrics:

The biometric authentication model is measured using evaluation measures such as False Acceptance Rate (FAR), False Rejection Rate (FRR) and Equal Error Rate (EER). Table 3.10 shows the performance metrics used in phase II.

Table 3.10: Performance metrics used in Phase II

S.no	Performance metrics	Description	Formula
1.	False Rejection Rate (FRR)	It measures the percentage ratio between incorrectly rejected authorized users against the total number of genuine users accessing the system. A lower false rejection rate means fewer rejects cases and legitimate users' easier access [32]	$\text{FRR} = \frac{\text{number of rejected genuine user}}{\text{total number of genuine user}}$
2.	False Acceptance Rate (FAR)	FAR measures the percentage ratio between falsely accepted unauthorized users against the total number of imposters accessing the system. Terms such as false match rate (FMR) or type 2 error refer to the same meaning. A smaller FAR indicates less imposter accepted [32].	$\text{FAR} = \frac{\text{number of accepted imposter}}{\text{total number of imposter}}$
3.	Equal Error Rate (EER)	EER is used to determine the biometric system accuracy. When FAR and FRR rates are equal, that intersection point is EER. The lower the value of EER, the higher the precision of the biometric system [32]	$\text{EER} = \frac{\text{number of authorized genuine user}}{\text{total number of users}}$

3.4.2 Experimental Analysis

This phase applies the CMU keystroke dataset as mentioned previously to perform keystroke based biometric authentication and OTP based secondary authentication.

The keystroke authentication is accomplished using the Gaussian Mixture Model (GMM) measure. The implementation is done in a jupyter notebook using python language. Table 3.11 explains the simulation parameters of GMM.

TABLE 3.11: SIMULATION PARAMETERS OF GMM

Parameters	Values
n_components	2
covariance_type	diag
tol	0.001
reg_covar	1e ⁻⁰⁶
max_iter	100
n_init	1
init_params	kmeans
weights_init	None
means_init	None
precisions_init	None
random_state	None
warm_start	False
verbose	0
verbose_interval	10

The Gaussian Mixture Model (GMM) model is trained using the train data to generate the particular user template. The model is tested using the test data to verify whether the particular user is genuine or imposter. The model is evaluated using EER, and obtained 19.8%, which is acceptable based on the fuzzy logic. Hence, the user is genuine in keystroke authentication and carries the secondary authentication.

In secondary authentication, randomly generated OTP. It is sent to the user for verification. Figure 3.11 shows the email received for OTP verification. Figure 3.12 shows the verification screen that verifies the user by questioning the OTP. If the randomly generated OTP is the same as when the user entered OTP, the verification is considered successful. Figure 3.13 depicts the successful verification of the user in the secondary authentication.

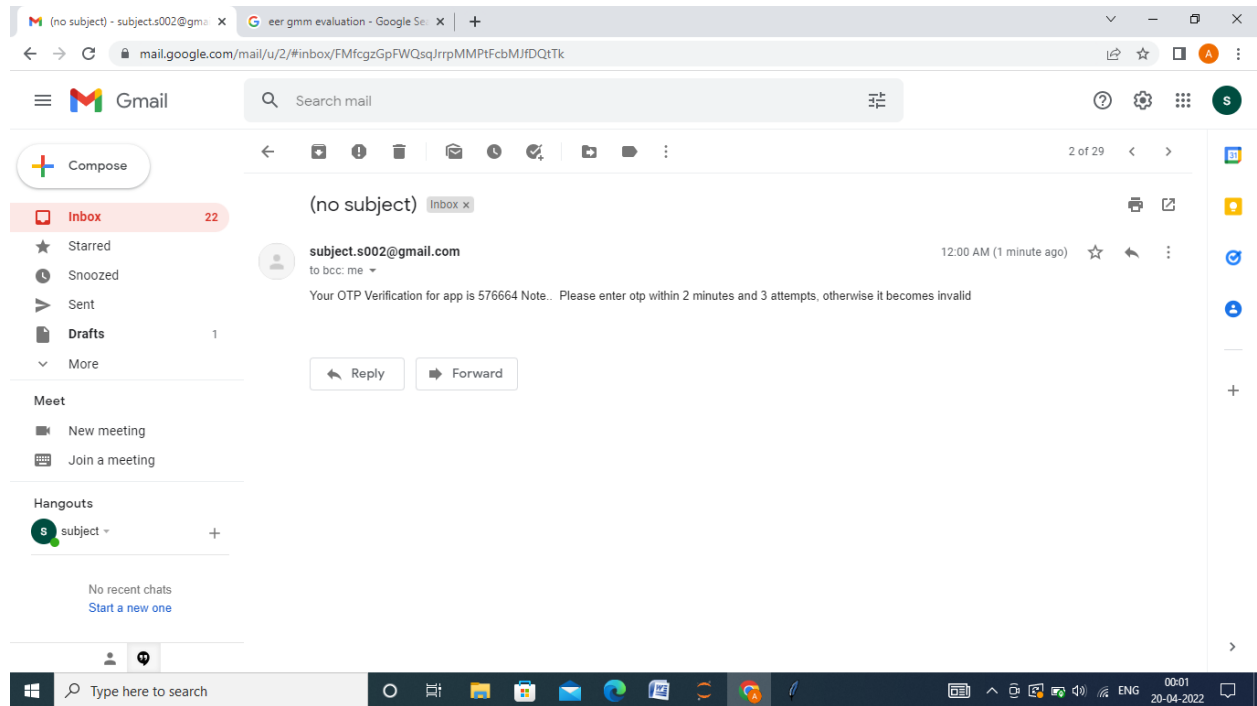


Figure 3.11: Received OTP verification in email

From the figure 3.11, it is observed that the email has been received by the user using the registered email address of user. The mail consists of OTP and the number of attempts to verify the authentication.

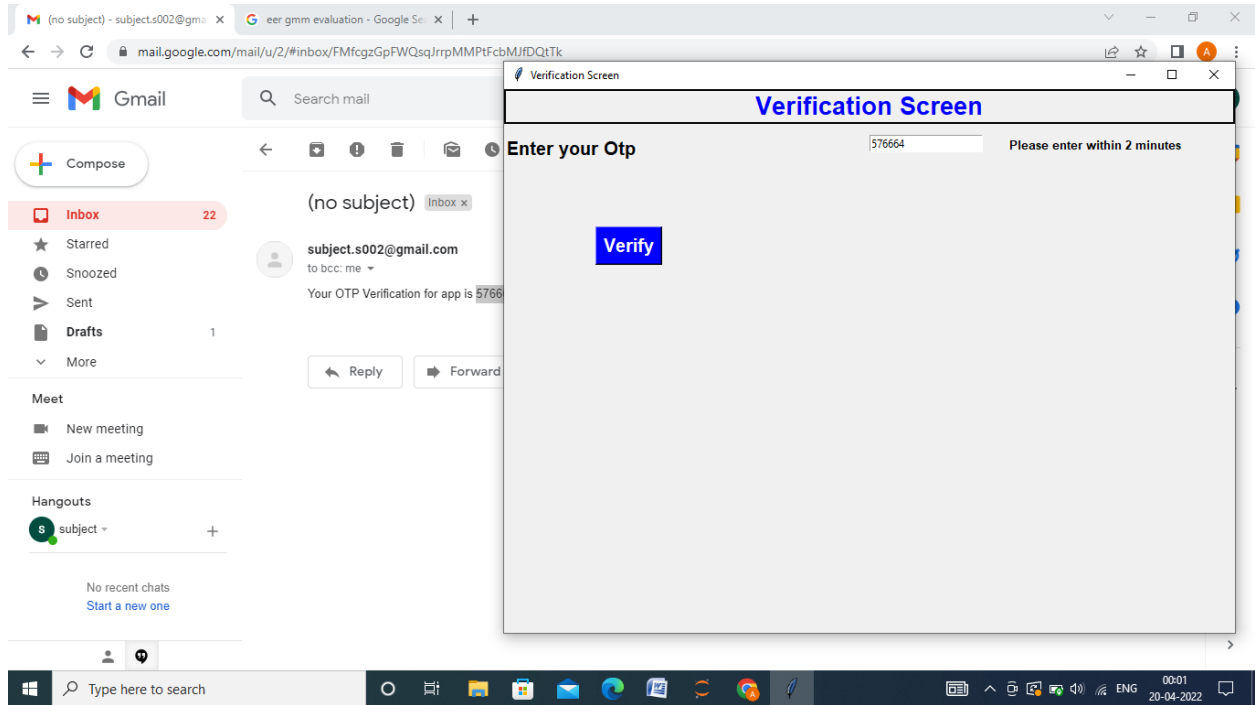


Figure 3.12: OTP in the Verification screen

From the figure 3.12, it is observed that the verification screen is used to validate the user via OTP. In the verification screen the user should enter the obtained OTP within the time limit and the number of attempts. After entering the OTP, the verify button is clicked.

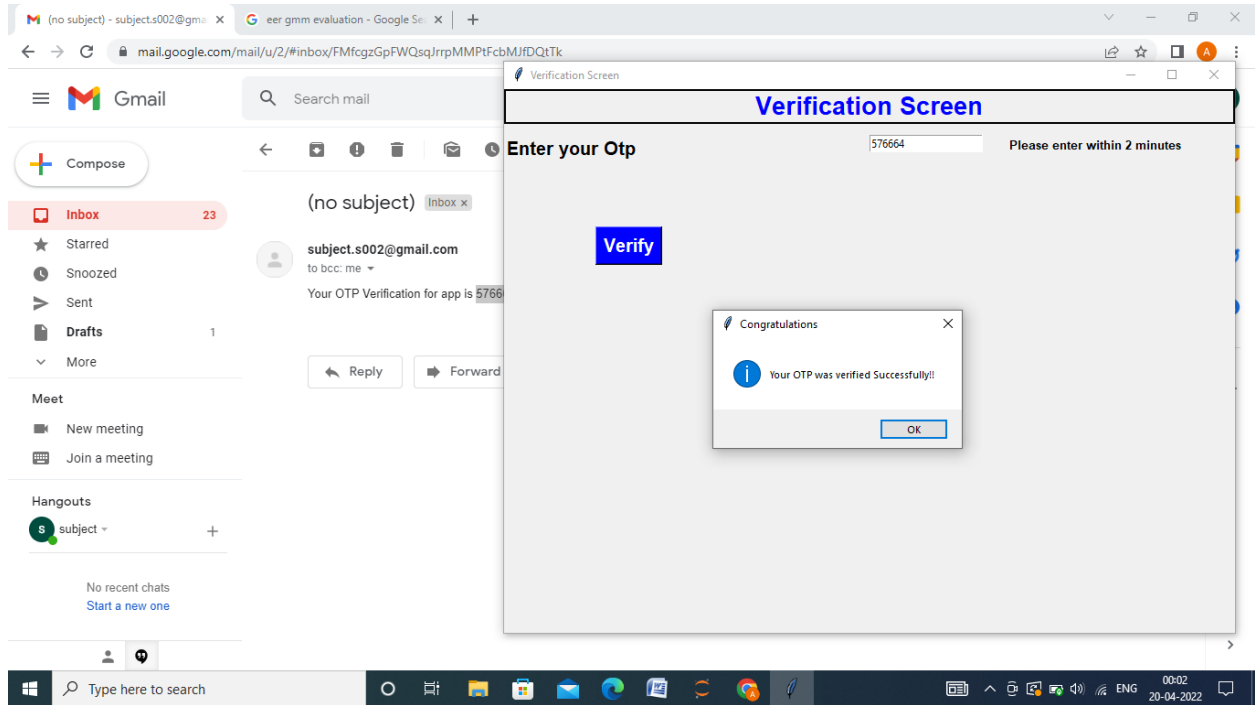


Figure 3.13: Result of OTP verification in the dialogue box

From the figure 3.13, it is observed that the result of the OTP verification is displayed in the dialogue box. This result is calculated after successful verification of user using OTP. If the user enters invalid OTP, the result is displayed as “Invalid OTP!!! Retry again”. The user got two OTP verification attempts. The user need to verify OTP within the number of limits for verification. If the number of verification succeeds the limit, the OTP verification of user is considered as “Unsuccessful OTP verification”.

The user has successfully verified using both keystroke based biometric authentication and OTP based secondary authentication. Based on the fuzzy logic in secondary authentication, it is understandable that the user is considered a genuine user.

SUMMARY

- 1) In this chapter, the proposed methodology for detection and mitigation of malicious insider threat is discussed with the experimental results.
- 2) The proposed methodology is able to detect and mitigate the malicious insider threat from cloud.
- 3) The next chapter concludes the work.

CONCLUSION

CHAPTER 4

CONCLUSION

The proposed detection phase is implemented using the CERT dataset in this research. It is processed, transformed and balanced using the pre-processing techniques are Data Integration, Data Transformation and Data level sampling is done using NearMiss-2 undersampling technique. Supervised machine learning technique, namely OCSVM, is used to train, classify and predict the balanced dataset for detecting the malicious insider activity using anomaly score. The performance metrics such as True Detection Rate and False Detection Rate are calculated to evaluate the performance of a supervised learning-based anomaly detection framework. The employee who possessed malicious activity in the top 50% is CCH0959 and CSF0929 and performed 242 and 26 malicious activity. Hence, the proposed framework achieved the maximum true detection rate with a zero percent false detection rate for detecting malicious insider threats in the cloud and making it suitable for real-world implementation. The detected malicious insider is further processed by applying keystroke-based biometric and OTP-based secondary authentication to mitigate the malicious insider threat.

The proposed mitigation phase is implemented using the CMU keystroke benchmark dataset. It is split into train and test data. The GMM model is used to train and test the data. The biometric authentication model is evaluated using FAR, FRR and EER. When the user obtained 19% EER with lower FAR and maximum FRR, the user is labelled as "Genuine" and eligible for secondary authentication. The user is considered "Genuine" in secondary authentication because the user satisfies the OTP verification. Since both the authentication is successfully verified, the user is considered "Genuine". While the unsuccessful verification in either authentication results in the user as "malicious". The malicious user can undergo a password change or temporary suspension of the user count. Hence, the proposed framework for detection and mitigation of malicious insider threat is successfully implemented.

FUTURE ENHANCEMENT

CHAPTER 5

FUTURE ENHANCEMENT

There are some of the short comings in detection and mitigation of malicious insider threat in cloud environment. Firstly, the deep learning techniques can be explored for best performance shortly in near future to detect the malicious insider threat. Secondly, the alternative existing available secondary authentication technique can be applied to enhance the mitigation strategies.

ACKNOWLEDGEMENT

ACKNOWLEDGEMENT

This work is supported in part by the Centre for Cyber Intelligence, DST CURIE AI Phase II Project, Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore, Tamilnadu, India.

REFERENCES

REFERENCES

- [1] B. Lindauer, J. Glasser, M. Rosen and K. Wallnau, "Generating Test Data for Insider Threat Detectors," *J. Wirel. Mob. Networks Ubiquitous Comput. Dependable Appl.* 2014, 5, pp. 80-94.
- [2] A. Gosain and S. Sardana, "Handling class imbalance problem using oversampling techniques: A review," in *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 2017.
- [3] D.J. Dittman, T. M. Khoshgoftaar, R. Wald, A. Napolitano, "Comparison of data sampling approaches for imbalanced bioinformatics data," in *The twenty-seventh international FLAIRS conference*, pp. 268-271, 2014.
- [4] N. Junsomboon and T. Phientrakul, "Combining over-sampling and under-sampling techniques for imbalance dataset," in *Proceedings of the 9th International Conference on Machine Learning and Computing - ICMLC 2017*, 2017.
- [5] T. Hasanin and T. Khoshgoftaar, "The effects of random undersampling with simulated class imbalance for big data," in *2018 IEEE International Conference on Information Reuse and Integration (IRI)*, 2018.
- [6] H. He, Y. Bai, E. A. Garcia, and S. Li, "ADASYN: Adaptive synthetic sampling approach for imbalanced learning," in *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, 2008.
- [7] B. W. Yap, K. A. Rani, H. A. A. Rahman, S. Fong, Z. Khairudin, and N. N. Abdullah, "An application of oversampling, undersampling, bagging and boosting in handling imbalanced datasets," in *Lecture Notes in Electrical Engineering*, Singapore: Springer Singapore, 2014, pp. 13–22.

- [8] K. Fujiwara et al., "Over- and under-sampling approach for extremely imbalanced and small minority data problem in health record analysis," *Front. Public Health*, vol. 8, p. 178, 2020.
- [9] C. Bunkhumpornpat and S. Subpaiboonkit, "Safe level graph for synthetic minority over-sampling techniques," in 2013 13th International Symposium on Communications and Information Technologies (ISCIT), 2013.
- [10] L. Abdi and S. Hashemi, "To combat multi-class imbalanced problems by means of over-sampling and boosting techniques," *Soft Comput.*, vol. 19, no. 12, pp. 3369–3385, 2015.
- [11] E. At, A. M, A.-M. F, and S. M, "Classification of imbalance data using Tomek link (T-link) combined with random under-sampling (RUS) as a data reduction method," *Glob. J. Technol. Optim.*, vol. 01, no. S1, 2016.
- [12] W. Jiang, Y. Tian, W. Liu and W. Liu. "An Insider Threat Detection Method Based on User Behavior Analysis," In *International Conference on Intelligent Information Processing*, pp. 421-429. Springer, Cham.
- [13] W. Eberle and L. Holder. "Applying graph-based anomaly detection approaches to the discovery of insider threats," In *2009 IEEE International Conference on Intelligence and Security Informatics*, pp. 206-208. IEEE.
- [14] L. Liu, O. De Vel, C. Chen, J. Zhang and Y. Xiang. "Anomaly-based insider threat detection using deep autoencoders," In *2018 IEEE International Conference on Data Mining Workshops (ICDMW)*, pp. 39-48. IEEE.
- [15] A. Diop, N. Emad, T. Winter and M. Hilia. "Design of an Ensemble Learning Behavior Anomaly Detection Framework," *International Journal of Computer and Information Engineering* 13, 10: 547-555.
- [16] J. Jiang, J. Chen, T. Guet et al. "Anomaly detection with graph convolutional networks for insider threat and fraud detection," In *MILCOM 2019-2019 IEEE Military Communications Conference (MILCOM)*, pp. 109-114. IEEE.
- [17] J. Kim, M. Park, H. Kim, S. Cho and P. Kang. "Insider threat detection based on user behavior modeling and anomaly detection algorithms," *Applied Sciences* 9, pp. 4018.

- [18] T.E. Senator, H.G. Goldberg, A. Memory et al. "Detecting insider threats in a real corporate database of computer usage activity," In Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 1393-1401.
- [19] Q. Lv, Y. Wang, L. Wang and D. Wang. "Towards a user and role-based behavior analysis method for insider threat detection," In 2018 international conference on network infrastructure and digital content (IC-NIDC), pp. 6-10. IEEE.
- [20] A. Gamachchi, L. Sun and S. Boztas. "A graph-based framework for malicious insider threat detection," arXiv preprint arXiv:1809.00141.
- [21] L. Liu, C. Chen, J. Zhang, O. De Veland Y. Xiang. "Doc2vec-based insider threat detection through behaviour analysis of multi-source security logs," In 2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), pp. 301-309. IEEE.
- [22] D.C. Le and N. Zincir-Heywood. "Anomaly detection for insider threats using unsupervised ensembles," IEEE Transactions on Network and Service Management 18, 2: 1152-1164.
- [23] P.A. Legg. "Visualizing the insider threat: challenges and tools for identifying malicious user activity," In 2015 IEEE Symposium on Visualization for Cyber Security (VizSec), pp. 1-7. IEEE.
- [24] P. Singh, V. Ranga. "Attack and intrusion detection in cloud computing using an ensemble learning approach," In 2021 Int. j. inf. tecnol. 13, pp. 565-571.
- [25] B. Sumitra, C. R. Pethuru, and M. Misbahuddin. "A survey of cloud authentication attacks and solution approaches," Int. J. Innov. Res. Comput. Commun. Eng, vol. 2, no. 10, pp. 6245-6253, 2014.
- [26] Tewari, A., & Verma, P. (2022). An Improved User Identification based on Keystroke-Dynamics and Transfer Learning. *Webology*, 19(1).
- [27] P. Maharjan, K. Shrestha, T. Bhatta, H. Cho, C. Park, M. Salauddin, M. T. Rahman, S. M. S. Rana, S. Lee, and J. Y. Park, "Keystroke dynamics based hybrid nanogenerators for biometric

authentication and identification using artificial intelligence,” *Advanced Science*, vol. 8, no. 15, p. 2100711, 2021.

[28] Sae-Bae, N., & Memon, N. (2022). Distinguishability of keystroke dynamic template. *PLOS ONE*, 17(1), e0261291. doi: 10.1371/journal.pone.0261291.

[29] A. Thapliyal, O. P. Verma, and A. Kumar, “Behavioral biometric based personal authentication in feature phones,” *Int. J. Electr. Comput. Eng. (IJECE)*, vol. 12, no. 1, p. 802, 2022.

[30] A. Huang, S. Gao, J. Chen, L. Xu, and A. Nathan, “High security user authentication enabled by piezoelectric keystroke dynamics and machine learning,” *IEEE Sens. J.*, vol. 20, no. 21, pp. 13037–13046, 2020.

[31] P. Baynath, K. M. Sunjiv Soyjaudah, and M. Heenaye-Mamode Khan, “Machine learning algorithm on keystroke dynamics pattern,” in *2018 IEEE Conference on Systems, Process and Control (ICSPC)*, 2018.

[32] C. Shi, J. Liu, H. Liu, and Y. Chen, “WiFi-enabled user authentication through deep learning in daily activities,” *ACM Trans. Internet Things*, vol. 2, no. 2, pp. 1–25, 2021.

[33] B. Bhana and S. Flowerday, “Passphrase and keystroke dynamics authentication: Usable security,” *Comput. Secur.*, vol. 96, no. 101925, pp. 1-13, 2020.

[34] A. Abo-alian, N. L. Badr, and M. F. Tolba, “Keystroke dynamics-based user authentication service for cloud computing: Keystroke Authentication Service,” *Concurr. Comput.*, vol. 28, no. 9, pp. 2567–2585, 2016.

[35] M. B. Bondada and M. S. B. S., “Analyzing user behavior using keystroke dynamics to protect cloud from malicious insiders,” in *2014 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM)*, 2014.

[36] Z. Chen, H. Cai, L. Jiang, W. Y. Zou, W. Zhu, and X. Fei, “Keystroke Dynamics Based User Authentication and its application in online examination,” in *2021 IEEE 24th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*), pp. 649–654, 2021.

- [37] C. Jadhav, S. Kulkarni, S. Shelar, K. Shinde, and N. V. Dharwadkar, "Biometric authentication using Keystroke Dynamics," 2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), pp. 870–875, 2017.
- [38] L. Aversano, M. L. Bernardi, M. Cimitile, and R. Pecori, "Continuous authentication using deep neural networks ensemble on keystroke dynamics," *PeerJ Comput. Sci.*, vol. 7, no. e525, pp. 1-27, 2021.
- [39] H. C. Chang, J. Li, C.-S. Wu, and M. Stamp, "Machine learning and deep learning for fixed-text keystroke dynamics," *arXiv [cs.LG]*, 2021.
- [40] A. Pentel, "Predicting age and gender by keystroke dynamics and mouse patterns," in *Adjunct Publication of the 25th Conference on User Modeling, Adaptation and Personalization - UMAP '17*, pp. 381-385, 2017.
- [41] S. Krishnamoorthy, L. Rueda, S. Saad, and H. Elmiligi, "Identification of user behavioral biometrics for authentication using keystroke dynamics and machine learning," in *Proceedings of the 2018 2nd International Conference on Biometric Engineering and Applications - ICBEA '18*, pp. 50-57, 2018.
- [42] I. Lamiche, G. Bin, Y. Jing, Z. Yu, and A. Hadid, "A continuous smartphone authentication method based on gait patterns and keystroke dynamics," *J. Ambient Intell. Humaniz. Comput.*, vol. 10, no. 11, pp. 4417–4430, 2019.
- [43] D. Shanmugapriya, and G. Padmavathi, "An efficient feature selection technique for user authentication using keystroke dynamics," *IJCSNS International Journal of Computer Science and Network Security*, vol. 11, no. 10, pp.191-195, 2011.
- [44] C. Giuffrida, K. Majdanik, M. Conti, and H. Bos, "I sensed it was you: Authenticating mobile users with sensor-enhanced keystroke dynamics," in *Detection of Intrusions and Malware, and Vulnerability Assessment*, Cham: Springer International Publishing, 2014, pp. 92–111.

[45] A. Huang, S. Gao, J. Chen, L. Xu, and A. Nathan, "High security user authentication enabled by piezoelectric keystroke dynamics and machine learning," *IEEE Sensors Journal*, vol. 20, no. 21, pp. 13037–13046, 2020.

[46] E. Schultz. "A framework for understanding and predicting insider attacks, " *Comput. Secur.* 2012, 21, pp.526–531.

[47] J. Glasser and B. Lindauer. "Bridging the gap: A pragmatic approach to generating insider threat data, " *In Proceedings of the 2013 IEEE Security and Privacy Workshops*, San Francisco, CA, USA, 23–24 May 2013; pp. 98–104.

[48] Geekflare Editorial. (2020, January 18). How to Prevent the Top 11 Threats in Cloud Computing? *Geekflare*. <https://geekflare.com/prevent-cloud-computing-threats/>

LIST OF PUBLICATIONS

LIST OF PUBLICATIONS

1) G. Padmavathi, D. Shanmugapriya and S. Asha, "A Framework to Detect the Malicious Insider Threat in Cloud Environment using Supervised Learning Methods," 2022 9th International Conference on Computing for Sustainable Global Development (INDIACom), 2022, pp. 354-358, doi: 10.23919/INDIACom54597.2022.9763205.

2) Padmavathi, G., Shanmugapriya, D., Asha, S. (2022). A Framework for Improving the Accuracy with Different Sampling Techniques for Detection of Malicious Insider Threat in Cloud. In: Uddin, M.S., Jamwal, P.K., Bansal, J.C. (eds) Proceedings of International Joint Conference on Advances in Computational Intelligence. Algorithms for Intelligent Systems. Springer, Singapore. https://doi.org/10.1007/978-981-19-0332-8_36

Proof for Scopus indexed IEEE conference paper

The screenshot shows the Scopus author profile for Shanmugapriya, D. The profile includes the Scopus logo, navigation links (Search, Lists, Sources, Scival), and buttons for 'Create account' and 'Sign in'. The author's name is 'Shanmugapriya, D.' and the profile is generated by Scopus Learn more. The author's affiliation is 'Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore, India'. The Scopus ID is 36618101800 and the ORCID ID is <https://orcid.org/0000-0002-7446-6749>. The profile also shows options to 'Edit profile', 'Set alert', 'Potential author matches', and 'Export to SciVal'. The 'Metrics overview' section displays 13 Documents by author, 37 Citations by 37 documents, and an h-index of 4. The 'Document & citation trends' section shows a line graph of Documents (blue bars) and Citations (black line) from 2010 to 2022. The 'Most contributed Topics' section is currently empty. At the bottom, there are statistics: 13 Documents, Cited by 37 Documents, 0 Preprints, 8 Co-Authors, Topics, and 0 Awarded Grants.

Conference Paper

A Framework to Detect the Malicious Insider Threat in Cloud Environment using Supervised Learning Methods

0

Padmavathi, G., Shanmugapriya, D., Asha, S.

Proceedings of the 2022 9th International Conference on Computing for Sustainable Global Development, INDIACom 2022, 2022, pp. 354–358

Citations

[Show abstract](#) [View at Publisher](#) [Related documents](#)

Proof for Conference proceeding published in Springer (Web of Science Indexed)



Conference proceedings | © 2022

Proceedings of International Joint Conference on Advances in Computational Intelligence IJCACI 2021

Editors: ([view affiliations](#)) Mohammad Shorif Uddin, Prashant Kumar Jamwal, Jagdish Chand Bansal

Presents research works in the field of computational intelligence

Provides original works presented at IJCACI 2021 held online during October 23–24, 2021

Serves as a reference for researchers and practitioners in academia and industry

Part of the book series: [Algorithms for Intelligent Systems](#) (AIS)

994 Accesses |  [Altmetric](#)

▼ eBook EUR 213.99

Price includes VAT (India)

- ISBN: 978-981-19-0332-8
- Instant PDF download
- Readable on all devices
- Own it forever
- Exclusive offer for individuals only
- Tax calculation will be finalised during checkout

[Buy eBook](#)

▶ Hardcover Book EUR 249.99

A Framework for Improving the Accuracy with Different Sampling Techniques for Detection of Malicious Insider Threat in Cloud

G. Padmavathi, D. Shanmugapriya, S. Asha
Pages 485–494

A Framework to Detect the Malicious Insider Threat in Cloud Environment using Supervised Learning Methods

G. Padmavathi

Department of Computer Science
Avinashilingam Institute for Home Science
and Higher Education for Women
Tamilnadu, India
padmavathi_cs@avinuty.ac.in

D. Shanmugapriya

Department of Computer Science
Avinashilingam Institute for Home Science
and Higher Education for Women
Tamilnadu, India
shanmugapriya_it@avinuty.ac.in

S. Asha

Department of Computer Science
Avinashilingam Institute for Home Science
and Higher Education for Women
Tamilnadu, India
20phcsf005@avinuty.ac.in

Abstract— A malicious insider threat is more vulnerable to an organization. It is necessary to detect the malicious insider because of its huge impact to an organization. The occurrence of a malicious insider threat is less but quite destructive. So, the major focus of this paper is to detect the malicious insider threat in an organization. The traditional insider threat detection algorithm is not suitable for real time insider threat detection. A supervised learning-based anomaly detection technique is used to classify, predict and detect the malicious and non-malicious activity based on highest level of anomaly score. In this paper, a framework is proposed to detect the malicious insider threat using supervised learning-based anomaly detection. It is used to detect the malicious insider threat activity using One-Class Support Vector Machine (OCSVM). The experimental results shows that the proposed framework using OCSVM performs well and detects the malicious insider who obtain huge anomaly score than a normal user.

Keywords—anomaly detection, behavioral model, insider threat detection, machine learning, OCSVM.

I. INTRODUCTION

Cloud computing is a framework that provide infrastructure, platform and software as a service to a wide range user with a metered cost. There are some security threats in cloud that can be handled using built-in security mechanism. But, it fails to handle the more destructible passive attack such as malicious inside threat. Since the malicious insider threat is more critical, it is crucial to detect the malicious insider threat in an organization. Previous research focuses on detecting the malicious insider in third party based. and few studies have addressed methods to detect insider threats. Machine learning provides sufficient algorithm for detection of malicious insider threat in cloud. It aims at developing a method to automatically identify users who perform unusual activities among all users without prior knowledge or rules [1]. Using supervised learning methods, the hidden pattern for each user is generated to further classify and predict the malicious insider threat in an organization.

In this research paper, the insider threat detection framework considers the user log details that consist of web activity, device connectivity and login details. The activity

logs are further trained using supervised learning-based anomaly detection framework using OCSVM.

OCSVM is used to classify and predict the normal and malicious activity whereas the particular user activity scores highest anomaly score than threshold value is considered as malicious activity. The user who executes malicious activity is considered as malicious insider.

The rest of this paper is organized as follows: In section II, the related study for insider threat detection framework is reviewed. In section III, the novel supervised learning-based anomaly detection framework is explained in detail. Section IV reports the experimental results and observations. In section V, the research work is concluded with the interesting future research ideas.

II. RELATED WORK

Table I describes the different studies conducted in the field of various anomaly detection framework.

Based on the reviewed literature, summarized in Table I, it is observed that different types of machine learning algorithms are applied to detect the malicious insider. The reviewed studies lack detection accuracies. Therefore, the reviewed studies laid the foundation to develop and implement supervised machine learning based anomaly detection technique to improve the detection of a malicious insider in an organization.

III. PROPOSED FRAMEWORK FOR MALICIOUS INSIDER DETECTION

Figure 1 shows the steps of the proposed framework, using supervised learning based anomaly detection framework, to analyze and detect the malicious insider activity in an organization. The framework implements OCSVM. The proposed framework has three phases:

- Phase I demonstrates the dataset used.
- In Phase II, the pre-processing techniques to handle the imbalanced class problem are explained.
- In phase III, the supervised learning-based anomaly detection is implemented to detect the malicious insider threat.

TABLE I. SUMMARY OF RELATED WORK

Reference	Algorithms applied	Observations
[4]	XGBoost, SVM, Random Forest (RF)	User behaviour analysis using XGBoost outperforms other algorithms based on F-measure up to 99.96% to detect the malicious activity using CERT dataset .
[5]	GBAD-MDL, GBAD-P (probability) and GBAD-MPS (maximum partial substructure)	Graph-based anomaly detection using MDL algorithm identifies the graph-based anomalies such as email, phone traffic and business process to detect the insider threat than Probability and MPS algorithm.
[6]	Deep Autoencoder (AE)	Deep A.E. detects all malicious insider activity with a reasonable false positive rate using US-CERT data.
[7]	IForest, One-Class SVM, Local outlier factor (LOF), Elliptic envelope (EE), artificial neural network (ANN), Gaussian naive Bayes(Gnb), Bagging classifiers (Bgc), random forest (RF) and gradient boosting (Gbc)	Ensemble learning behavior using Gbc algorithm outperforms other algorithms with (75%-99%) in both unsupervised learning based testing and supervised learning based testing. An ANN followed this with (60%-99%) result in both tests.
[8]	RF, SVM, Logistic Regression (LR), Convolutional Neural Network (CNN), Graph Convolutional Network (GCN)	GCN performs better than other algorithm (based on accuracy, precision and recall) to detect malicious insider and fraud activities.
[9]	Gaussian density estimation, Parzen window density, Principal component	User behavior modelling and anomaly detection using Parzen and PCA provided a better result than other algorithms to detect malicious insider threats.
[10]	IP Thief Ambitious Leader Scenario Detector, File Events Indicator Anomaly Detection, Relational Pseudo Anomaly Detection, Repeated Impossible Discrimination Ensemble, Grid-based Fast Anomaly Discovery given Duplicates (GFADD)	The multiple methods detect the malicious insider threat using computer log activity in an actual corporate database.
[11]	Isolation Forest	MURB outperforms the ADAD with 80% precision and accuracy for detection of the malicious insider threat using CERT data.
[12]	Isolation Forest	The combined graph-based anomaly detection framework identifies 79% of individuals as Genuine users and 31% as malicious insiders with suspicious activity.
[13]	Behaviour analysis	The new behaviour analysis framework named Doc2vec simplifies insider threat detection based on spatial and temporal metrics.
[14]	AutoEncoder, Isolation Forest, Lightweight on-line detector of anomalies (LODA), Local Outlier Factor (LOF)	Unsupervised ensemble-based anomaly detection using Autoencoder outperforms the other algorithm based on voting metrics to detect the malicious insider threat.
[15]	Visual Analytics	Visual analytics is recommended to detect malicious insider threat activity based on profiling behaviour and selected features as a mitigation strategy.
[16]	Boosted tree, bagged tree, subspace discriminant and RUSBoost	It achieves 97.24% accuracy to detect the intruders in cloud environment and it outperforms the other existing techniques.

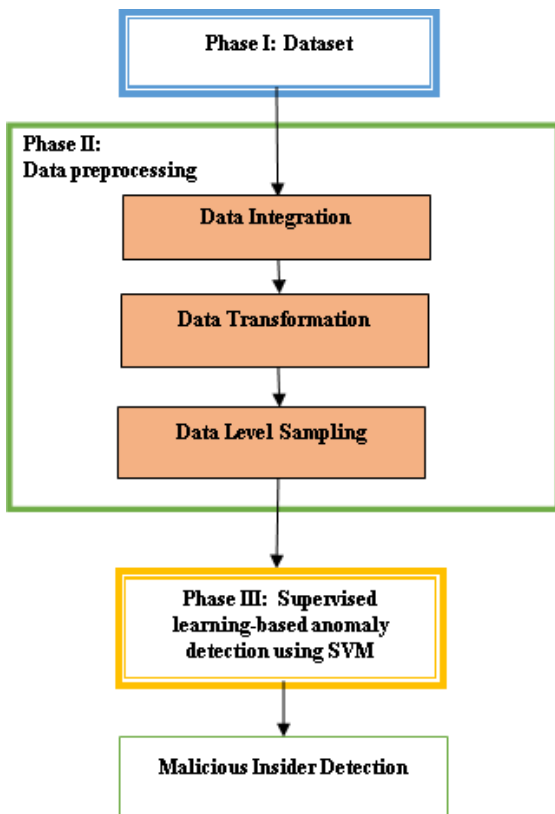


Fig. 1. Malicious Insider Detection Framework

A. Dataset

To perform the above-mentioned framework, “CERT Insider Threat Tools” dataset (Carnegie Mellon’s Software Engineering Institute, Pittsburgh, PA, USA) [20] is used. The CERT dataset is an artificially generated synthetic dataset for the purpose of validating insider-threat detection frameworks [1]. They are (i) employee activity log in computer such as logon, device, http, file and email. The activity information comprises of ID, timestamps, user ID, pc ID and activities. (ii) organization structure information such as employee departments and roles. The US-CERT dataset has six major versions (R1 to R6) [1]. The version R3 has two distinctions. They are R3.1 and R3.2. It includes the information of activity, employees and malicious insider activities based on dataset variation. In this research, version R3.1 is considered as baseline. It includes 1000 users, amongst two employees performed maliciously. The log information that satisfies the selected scenarios are http, logon and device. It contains the information such as unique id, date, user id, pc id, activity. In LDAP folder, the information includes employee name, user id, email, role, project, business unit, functional unit, department, team and supervisor.

B. Data Pre-processing

In the selected dataset, the activities of employees are warehoused in three tables such as logon, device and HTTP. The different table is required to combined as a single

homogeneous employee activity table. It is accomplished using pre-processing techniques such as data integration, data transformation and data level sampling. These pre-processing techniques enhances the data making it suitable for insider detection framework.

The integrated data undergoes the data transformation method using categorical encoding to further transform the data into numerical value.

Since the transformed data have huge number of non-malicious class instance than the instance of malicious class, it encounters the imbalanced class problem. The training of imbalance class instance will cause the insignificant result in detection of the malicious insider threat in an organization. To overcome the imbalanced class problem, data level sampling technique is utilized. it consists of undersampling and oversampling technique. Nearmiss-2 is one of the suitable undersampling technique to handle the imbalanced class data. Table II illustrates the training set instance before and after applying undersampling technique.

TABLE II. TRAINING SET INSTACE BEFORE AND AFTER UNDERSAMPLING TECHNIQUE

Training Set	Before Sampling	After Sampling
Majority class instance	39732	268
Minority class instance	268	268

C. Supervised Learning based Anomaly Detection

The sampled data is well balanced to perform the insider threat anomaly detection. In the insider detection phase, OCSVM is used to train the balanced data. it is accomplished using supervised learning-based anomaly detection using OCSVM. The OCSVM classifier input the sampled data to predict the malicious scores for each activity. In this paper, OCSVM is used to predict the anomaly score based on activity of the employee. The employee with highest anomaly score is considered as malicious insider.

IV. IMPLEMENTATION AND RESULTS

The proposed approach uses CERT dataset to perform data pre-processing and anomaly detection using OCSVM classifier. The implementation is done in Jupyter Notebook using python language. Table III lists the simulation parameters of OCSVM.

TABLE III. SIMULATION PARAMETERS OF OCSVM

Parameter	Value
Kernel	RBF
Gamma	0.001
Nu	0.02

The simulation process of the proposed novel methodology is previously explained and the result is explained below.

Figure 2 shows the sampled data consisting of genuine user activity and malicious activity in an organization. But the activity of genuine user is more less as compared to the activity of malicious user (Figure 3). It is required to detect the malicious insider in an organization. The supervised machine learning algorithm such as OCSVM is implemented

to train, classify and predict the malicious activity in an organization. the predicted value comprises of 1 and -1 where -1 indicates the malicious activity. Based on the activity of each employee in an organization, the anomalous score is calculated. By altering the threshold value, the anomalies are filtered.

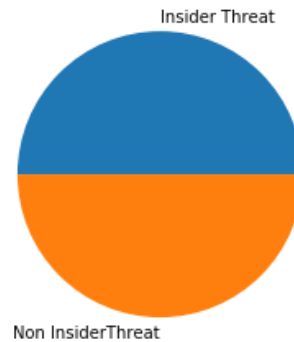


Fig. 2. Balanced data of Genuine user and malicious user

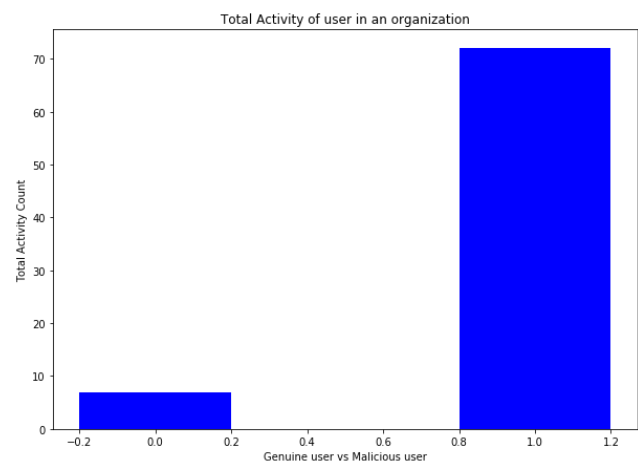


Fig. 3. Activity count of Genuine user and malicious user

Table IV presents the threshold value, total filtered anomalous activity, true detection rate and false detection rate.

TABLE IV. DESCRIPTION OF TOP X% OF MALICIOUS ACTIVITY

Top X%	Threshold value	Total filtered activity (out of 268)	True detection rate (in %)	False detection rate (in %)
10%	0.0195312524	54	20.14%	0%
20%	0.0195362745	106	39.55%	0%
30%	0.0200169149	161	60.07%	0%
40%	0.0202734374	214	79.05%	0%
50%	0.0202734375	268	100%	0%
60%	0.0202734404	321	100%	19.77%
70%	0.0203840269	369	100%	37.68%
80%	0.0204303234	405	100%	51.11%
90%	0.0204303234	405	100%	51.11%
99%	0.0205078125	526	100%	96.26%

In the top 10% and top 20% of malicious activity the true detection is more less. It detects 20% and 39% of malicious activity in an organization which is insignificant in detecting the crucial malicious insider. It detects the 60% to 79% of

malicious activity of employee in top 30% and 40% activity. In top 50%, all the malicious activity is detected and achieved 100% true detection rate. Since it doesn't detect the activity of genuine user as malicious activity it achieves 0% false detection rate. From top 60% to top 99%, the false detection rate increases abruptly while true detection rate remains same. Hence, to detect all the malicious activity in an organization with 0% false detection rate, top 50% with threshold value of 0.202734375 is implemented.

Table V shows the activity of malicious employee in top 50%. The employee namely CCH0959 performed the total number of 242 malicious activities in an organization. whereas the employee CSF0929 performed the total number of 26 malicious activities in an organization.

TABLE V. ACTIVITY OF MALICIOUS EMPLOYEE IN TOP 50%

Top 50%	User	Total No of Malicious Activities
	CCH0959	242
	CSF0929	26

Figure 4 demonstrates the detection of malicious insider based on employee activity in an organization. Figure 5 shows the employee labelled as malicious insider is correctly detected in an organization. Figure 6 depicts the anomalous activity of malicious insider is detected. It shows that all the malicious activity of insider has been detected without any false detection of genuine user.

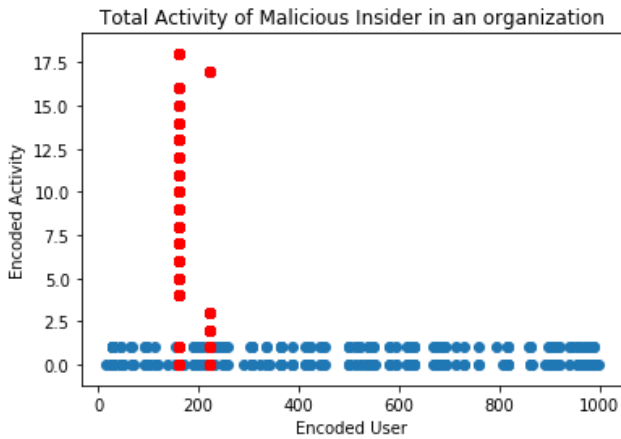


Fig. 4. Malicious insider based on employee activity

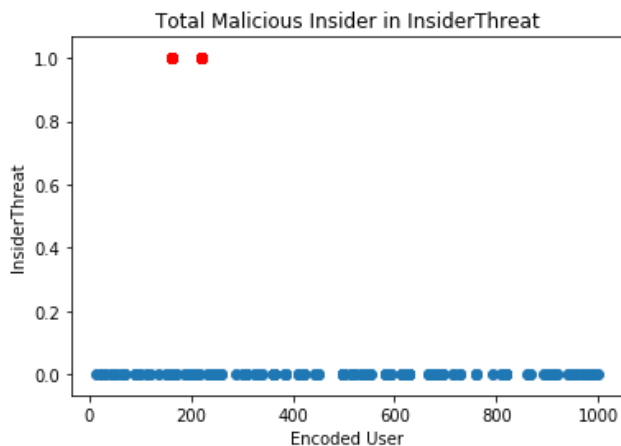


Fig. 5. Malicious insider detected in an organization

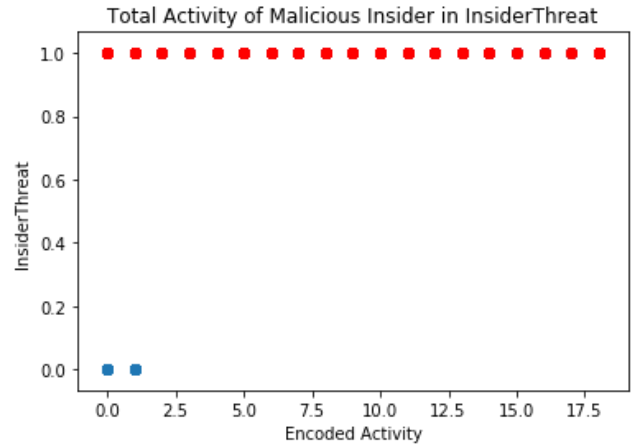


Fig. 6. Malicious activity of malicious employee in an organization

The proposed insider-detection model generated 20.14% true detection rate in top 10% of malicious activity with zero false detection rate. In top 30% and 40% of malicious activity, the true detection rate raised abruptly to 60.07% and 79.05% with no false detection rate. In top 50%, the proposed framework achieved 100% true detection rate with 0% false detection rate and satisfies the performance to detect the malicious insider threat in cloud environment.

V. CONCLUSION

In this paper, supervised learning-based anomaly detection model is implemented using CERT dataset. The dataset is processed, transformed and balanced using pre-processing techniques (data integration, data transformation and data level sampling). Supervised machine learning technique namely OCSVM is used to train, classify and predict the balanced dataset for detecting the malicious insider activity within organization using anomaly score. The performance metrics such as true detection rate and false detection rate is calculated to evaluate the performance of supervised learning-based anomaly detection framework. The employee who possessed malicious activity in top 50% is CCH0959 and CSF0929 and performed the total of 242 and 26 malicious activity within the organization. Hence, the proposed framework achieved the maximum true detection rate with zero percent false detection rate and making it suitable for real world implementation. In near future, deep learning techniques can be explored for best performance.

ACKNOWLEDGMENT

This work is partially supported by Centre for Cyber Intelligence, DST CURIE AI Phase II Project, Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore, Tamilnadu, India.

REFERENCES

- [1] B. Lindauer, J. Glasser, M. Rosen and K. Wallnau, “Generating Test Data for Insider Threat Detectors,” *J. Wirel. Mob. Networks Ubiquitous Comput. Dependable Appl.*, vol 5, pp.80-94, 2014.
- [2] E. Schultz. “A framework for understanding and predicting insider attacks,” *Comput. Secur.*, vol. 21, pp.526–531, 2012.
- [3] J. Glasser and B. Lindauer. “Bridging the gap: A pragmatic approach to generating insider threat data,” in *Proc. of the 2013 IEEE Security and Privacy Workshops*, San Francisco, CA, USA, 23–24 May 2013; pp. 98–104.

- [4] W. Jiang, Y. Tian, W. Liu and W. Liu. "An Insider Threat Detection Method Based on User Behavior Analysis," in Proc. of the International Conference on Intelligent Information Processing, 2018, pp. 421-429.
- [5] W. Eberle and L. Holder. "Applying graph-based anomaly detection approaches to the discovery of insider threats," in Proc. of the 2009 IEEE International Conference on Intelligence and Security Informatics, 2009, pp. 206-208.
- [6] L. Liu, O. De Vel, C. Chen, J. Zhang and Y. Xiang. "Anomaly-based insider threat detection using deep autoencoders," in Proc. of the 2018 IEEE International Conference on Data Mining Workshops (ICDMW), 2018, pp. 39-48.
- [7] A. Diop, N. Emad, T. Winter and M. Hilia. "Design of an Ensemble Learning Behavior Anomaly Detection Framework," International Journal of Computer and Information Engineering, vol. 13, no. 10, pp. 547-555, 2019.
- [8] J. Jiang, J. Chen, T. Gu et al. "Anomaly detection with graph convolutional networks for insider threat and fraud detection," in Proc. of the IEEE Military Communications Conference (MILCOM), 2019, pp. 109-114.
- [9] J. Kim, M. Park, H. Kim, S. Cho and P. Kang. "Insider threat detection based on user behavior modeling and anomaly detection algorithms," Applied Sciences, vol. 9, 2019.
- [10] T. E. Senator, H. G. Goldberg, A. Memory et al. "Detecting insider threats in a real corporate database of computer usage activity," in Proc. of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2013, pp. 1393-1401.
- [11] Q. Lv, Y. Wang, L. Wang and D. Wang. "Towards a user and role-based behavior analysis method for insider threat detection," in Proc. of the 2018 international conference on network infrastructure and digital content (IC-NIDC), 2018, pp. 6-10.
- [12] A. Gamachchi, L. Sun and S. Boztas. "A graph based framework for malicious insider threat detection," arXiv preprint arXiv:1809.00141.
- [13] L. Liu, C. Chen, J. Zhang, O. De Vel and Y. Xiang. "Doc2vec-based insider threat detection through behaviour analysis of multi-source security logs," in Proc. of the 2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), 2020, pp. 301-309.
- [14] D. C. Le and N. Zincir-Heywood. "Anomaly detection for insider threats using unsupervised ensembles," IEEE Transactions on Network and Service Management, vol. 18, no. 2, pp. 1152-1164, 2021.
- [15] P. A. Legg. "Visualizing the insider threat: challenges and tools for identifying malicious user activity," in Proc. of the 2015 IEEE Symposium on Visualization for Cyber Security (VizSec), 2015, pp. 1-7.
- [16] P. Singh and V. Ranga. "Attack and intrusion detection in cloud computing using an ensemble learning approach," International Journal of Information Technology, vol. 13, pp. 565-571, 2021.

Chapter 36

A Framework for Improving the Accuracy with Different Sampling Techniques for Detection of Malicious Insider Threat in Cloud



G. Padmavathi, D. Shanmugapriya, and S. Asha

1 Introduction

Storing of information and accessing of resources from anywhere is possible by anyone at any time by cloud. Many threats can attack the cloud, and one of the crucial threats is the malicious insider threat. A malicious insider is an individual who threatens to access confidential data and pretend to be a legitimate user within the organization. A malicious insider may cause data leakage leads to substantial financial and reputation loss. So, it is crucial to detect the malicious insider threat in an organization. Hence, a framework is proposed to detect a malicious insider threat. The real-world malicious insider data has been gathered from the US-Computer Emergency Response Team (CERT), which contains information regarding the malicious activity and non-malicious activity [1]. But non-malicious activity contains majority class instances, while malicious activity contains minority class instances. It is difficult to detect real malicious insider threats. The reason remains the same, which is possible when an instance of one class has maximum distribution than the other class instance. For example, in CERT data, the instance of non-malicious activity has a higher proportion than malicious activity. Whereas the instance of a non-malicious activity is considered as majority class, and the instance of malicious activity is regarded as a minority class. The classifier would consider the minority class as

G. Padmavathi · S. Asha (✉)

Department of Computer Science, Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore 641043, Tamilnadu, India
e-mail: 20phcsf005@avinuity.ac.in

G. Padmavathi

e-mail: padmavathi_cs@avinuity.ac.in

D. Shanmugapriya

Department of Information Technology, Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore 641043, Tamilnadu, India
e-mail: shanmugapriya_it@avinuity.ac.in

noise or outlier during the training, and misclassification occurs. In the classification technique, the Class Imbalance problem arises due to the inaccurate classification of the minority class [2]. It suppresses the performance of supervised classification algorithms.

The class imbalance problem will arise two critical issues in the supervised classification algorithm. They are (i) Misassumption and misclassification due to unequal proportion of class instance and (ii) Inaccurate prediction of the minority class, which will suppress the performance of the supervised classification algorithm due to inaccurate prediction of the minority class. Therefore, many techniques have been used to improve the accuracy and minimize the inaccurate prediction of the minority class. This paper utilizes different oversampling and undersampling techniques to solve the class imbalance problem and enhance the accurate prediction of malicious insider threats. The sampled data is trained using an SVM classifier to evaluate the performance of different sampling techniques using accuracy, f-score, precision and recall. The entire paper is organized into four sections. Section 2 tabulates the literature study in different sampling techniques. Section 3 explains the methodology overview. Section 4 illustrates the result and discussion. Section 5 concludes the research and suggests possible scope for future enhancement.

2 Background Study

The primary focus is to solve the class imbalance problem in imbalanced CERT data comprising malicious insider threats. The following Table 1 describes the work done in the area of various sampling techniques.

Table 1 Study of various sampling techniques

S. no	Author	Sampling techniques	Classification algorithm	Observations
1	Gosain and Sardana (2017)	SMOTE, ADASYN, Borderline-SMOTE, safe level-SMOTE	Naïve bayes, SVM and Nearest Neighbor	Safe Level SMOTE performed better than other oversampling techniques based on f-measure and g-mean [3]
2	Dittman et al. (2014)	RUS, ROS, SMOTE	KNN, SVM	RUS classified better than other techniques in SVM and KNN based on AUC-curve [4]

(continued)

Table 1 (continued)

S. no	Author	Sampling techniques	Classification algorithm	Observations
3	Junsomboon and Phienthrakul (2017)	Neighbor Cleaning Rule (NCL), SMOTE	Naïve Bayes, Sequential Minimal Optimization (SMO) and KNN	Combined NCL and SMOTE provided a better result than SMOTE, NCL and ordinary data in various classifiers based on recall measures [5]
4	Hasanin and Khoshgoftaar (2018)	RUS	Random Forest	The minority class between 0.1% to 1% true positive rate is outperformed than 10% and 100% of class balanced data [6]
5	Haibo He et al. (2008)	ADASYN and SMOTE	Decision tree	The ADASYN algorithm provided better accuracy than SMOTE [7]
6	Yap et al. (2014)	ROS, RUS, AdaBoost	Classification and Regression Tree (CART), C5 and Chi-Square Automatic Interaction Detection (CHAID)	RUS outperformed the other sampling techniques in three Decision Tree algorithm based on accuracy, sensitivity, specificity and precision [8]
7	Fujiwara et al. (2020)	ADASYN, SMOTE, AdaBoost, RUSBoost, hyperSURF, HUSBoost and proposed HUSDOS-Boost sampling	Random Forest	HUSDOS-Boost outperformed the RUSBoost and provided 0.69% of G-mean to detect stomach cancer with the minority class instance less than 30 [9]
8	Bunkhumpornpat and Subpaiboonkit (2013)	Improved SMOTE, Borderline-SMOTE and Safe-Level-SMOTE	Naive Bayes, Decision tree, KNN and RIPPER	Improved SMOTE provided a better result than other techniques on various classifiers and achieved 73% of F-measure and 78% of AUC [10]

(continued)

Table 1 (continued)

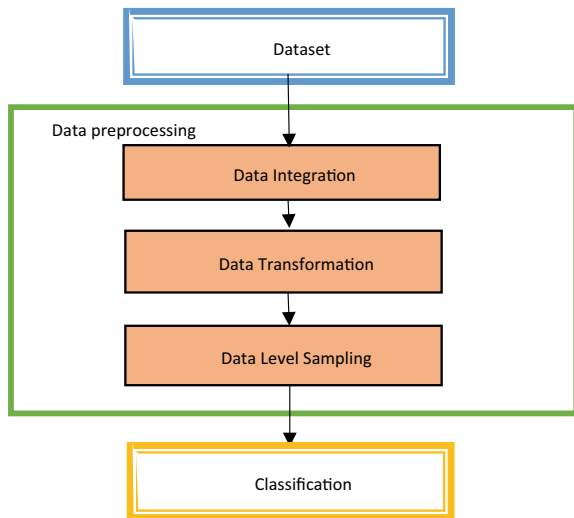
S. no	Author	Sampling techniques	Classification algorithm	Observations
9	Abdi and Hashemi (2015)	Mahalanobis Distance-Based Over-Sampling Technique (MDO), SMOTE, Borderline-SMOTE, and ADASYN	Decision Tree, KNN and RIPPER	MDO performed better than other techniques with various classifiers in terms of MAUC and precision [11]
10	Elhassan and Aljurf (2016)	Tomek’s Link(T-Link), RUS, ROS and SMOTE	SVM, ANN, Random Forest (RF) and Logistic Regression (LR)	T-Link performed best among various classifiers based on F-statistic, G-mean and AUC [12]

The above table shows that the different sampling techniques are applied to handle the class imbalance problem. Hence, the different sampling techniques are implemented and compared to improve the correct detection of a malicious insider in an organization.

3 Methodology

The Following Fig. 1 illustrates the proposed methodology of minority class classification with different sampling techniques to detect a malicious insider threat.

Fig. 1 Overview of proposed methodology



3.1 Dataset

The benchmark dataset is collected from the cyber security-based CERT [13] Division. The synthetic dataset based on malicious activity in the cloud environment has been collected. THE gathered US-CERT dataset consists of log details based on emails, web connection, device connectivity status, and login of malicious and non-malicious users. The dataset version r3.1 is considered as a primary dataset to analyze and detect the malicious insider threat. Some malicious insider threat-based scenarios [14] are defined below:

- Scenario 1: An individual in an organization working after working hours, often used to carry a removable drive and uploaded the important information to wikileaks.org. Later resigned from the organization.
- Scenario 2: An individual in an organization visited job websites and beseeched employment opportunities from a competitor of the business. The abnormal behaviour of the employee increases in data transfer using removable drives. Later resigned from the organization.
- Scenario 3: Unauthenticated or unsatisfactory system administrator tries to install malicious software to collect sensitive information and utilize the removable drives for data transmission from the particular authorized system. Gather sensitive information to access the authenticated system. It also contains emails regarding sensitive information in an unusual manner in an organization. Later, resigned from the organization.
- Scenario 4: Over three months, individuals frequently logged into other user's computers. Searched and forwarded files to a personal email address.
- Scenario 5: Uploaded documents to Dropbox for personal gain.

3.2 Data Pre-processing

The primary CERT data contains log details of 516 days, where 4000 users generate 135,117,169 log events [14]. The events are activities including email-based, login-based, device storage-based, HTTP operations, psychometric details, file information and daily log details. This paper considers scenario-1 and scenario-2 among the above mentioned five scenarios. So, the primary data related to selected scenarios are regarded as Base data, and others are neglected. The base data undergoes two pre-processing steps to make the data suitable for classification. It includes data integration and data transformation.

Data Integration. Detection of Malicious insider threat records related to device status, login status and HTTP operation satisfies the above-selected scenario. The selected records are integrated using simple feature concatenation techniques. While other records are neglected. The following Table 2 demonstrates the feature details of integrated data.

Table 2 Feature details of integrated data

Features	Description
InsiderThreat	It considers malicious activity or not
Vector	It is considered as the origin of data
Date	Date of the particular event
User	User id who carries particular activity
Pc	Unique identification for each computer
Activity	Action of particular user

Table 3 Transformed data

Features	Before transformation	After transformation
InsiderThreat	1	1
Vector	Logon	0
Date	07-01-2010 02:23:00	1,280,707,200
User	CCH0959	4
Pc	PC-0588	128
Activity	http://linkedin.com/jobs/displayhome.html	750

Data Transformation. The integrated data needs to be transformed to a categorical value for further processing. The features namely “vector”, “pc”, “user” and “activity” from integrated data converted into a numerical value. The value of “date” is converted into the number of epochs. The following Table 3 shows the details of transformed data.

Data Level Sampling. It is necessary to balance the instance in all classes for accurate classification. To solve the class imbalance problem, three different types of techniques have been used. They are Data level solution, Algorithmic level solution, Ensemble-based learning solution [15]. The solution at the data level for the class imbalance problem is based on sampling methods [16]. This technique provides the solution by altering the pattern of data distribution. It is also said to be restructuring the imbalanced class data to make it well-balanced data. It is accomplished by both undersampling and oversampling. The different types of oversampling techniques are Synthetic Minority Over-Sampling Technique (SMOTE), Adaptive Synthetic (ADASYN) and Random Oversampling (ROS). Some of the essential undersampling techniques are Edited Nearest Neighbours (ENN), Near-Miss 1 (NM-1), Near-Miss 2 (NM-2), Random Under sampler (RUS), Tomek-link (T-L) have been implemented.

In the pre-processed dataset, a feature like “InsiderThreat” is the target variable where the majority class instance “0” denotes non-malicious activity and minority class instance “1” indicates malicious activity. It is difficult to classify the minority class because the minority class instance is lesser than the majority class instance. So, a class imbalance arises during classification, where the data is distributed unequally for all the classes. This results in misclassification and misinterpretation of data. To

handle the class imbalance problem, data-level solutions such as oversampling and undersampling techniques are recommended.

Oversampling Techniques. The primary focus of Oversampling technique is to replicate the instance of minority class until the dataset is balanced. Since the size of minority class instances would increase abruptly, the learning time also increases. This paper considers the three oversampling algorithms to resample the imbalanced data. They are ROS, SMOTE and ADASYN. One of the common techniques in oversampling is ROS. It multiplies the instance of the minority class randomly by replicating the minority class instance. Thus, it raises the problem known as overfitting. To overcome the overfitting [4, 8, 12], artificial synthetic methods are recommended.

In SMOTE Eq. (1), a new artificial synthetic dataset is generated by combining minority class instance x_i and interpolation within KNN, namely x_{zi} [3, 4, 7, 9, 11, 12].

$$x_{new} = x_i + \lambda(x_{zi} - x_i) \quad (1)$$

where the λ is denoted as a random number between 0 and 1, the balanced data is created by interpolation between x_i and x_{zi} . Minority instances are generated using (i) Regular. (ii) Borderline approach using KNN (iii) SVM approach [10]. SMOTE modifies the artificial instance of minority class based on weight for each class is called ADASYN. It generates several instances for minority classes proportional to the number of the adjacent class instance [3, 7–10]. It concentrates on outlier or minority class instances.

Undersampling Technique. The primary focus of the Undersampling technique is to eliminate the instance of the majority class until the dataset is balanced. The decrease in the size of the majority class instance decreases the learning time [6]. This paper focuses on five undersampling algorithms that are used to balance the class imbalanced data. They are RUS, NM-1, NM-2, T-L and ENN.

One of the common techniques in undersampling is ROS. It minimizes the instance of the majority class in a random pattern until the majority class instance equals the minority class instance [6, 8, 10, 12]. Hence it causes loss of important information in the majority class. The idea of Near-miss is to resample the instance of the majority class necessary to differentiate all classes. In NM-1, the majority class instance is selected if it satisfies the minimum average distance for N neighbouring minority class instance. In NM-2, the majority class instance is chosen if it meets the minimum average distance for N outermost minority class instance. T-L's objective [12] is to clean the majority instance by eliminating the outlier same as a classifier.

$$d(x, z) < d(x, y) \text{ or } d(y, z) < d(x, y) \quad (2)$$

where d is defined as the distance between two instances. The link exists if the two instances of distinct classes are nearby each other. In ENN [11], the KNN eliminates the instance that fails to satisfy the neighbor.

3.3 Classification

The supervised classification technique such as the SVM classifier [3, 4, 12] is implemented using balanced data to accomplish the detection of malicious insider threats in an organization.

4 Results and Discussions

The following metrics are used to evaluate the proposed methodology. They are Accuracy, Sensitivity, Precision, F-score, True positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN). Accuracy is the most frequently used evaluation metrics and is defined as the proportion of accurately predicted instances to the overall sum of instances. It is expressed as follows:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (3)$$

Sensitivity or Recall is defined as a correctly predicted positive instance. It is also known as a measure of correctness. It is expressed as follows:

$$Sensitivity \text{ or } Recall = \frac{TP}{TP + FN} \quad (4)$$

Precision is a measure of exactness to predict the positive instance by eliminating the inaccurately predicted negative instance as a positive. It is expressed as follows:

$$Precision = \frac{TP}{TP + FP} \quad (5)$$

F-score is the weighted average of precision and recall in binary classification. The increase in precision and Recall increases the value of σ . It is expressed as follows:

$$F - score = \frac{2 * Precision * Recall}{2 * Recall + Precision} \quad (6)$$

The performance of the SVM classifier is compared using above mentioned evaluation metrics. The following Table 4 illustrates the performance of the SVM classifier after applying different oversampling and undersampling techniques. From Table 4, it is observed that the performance of ADASYN, ROS and SMOTE remains the same, and the Recall of a non-malicious event is less. Hence, the handling of imbalanced data using the oversampling technique is complex. In the undersampling technique, the f-score of NM-1 is almost more negligible, and specificity is unsatisfactory where it fails to detect the malicious activity. ENN and T-L achieved equal modest performance. NM-2 surpasses the ROS and improves the generation of an artificial minority class instance dramatically. Thus, it achieves high performance in Recall, Precision, F-score and Accuracy.

Table 4 Performance metrics of eight sampling methods

	Accuracy	F score	Precision	Recall
<i>Oversampling techniques</i>				
ADASYN	0.680375	0.80 ± 0.03	0.99 ± 0.02	0.67 ± 0.77
ROS	0.680375	0.80 ± 0.03	0.99 ± 0.02	0.67 ± 0.77
SMOTE	0.680375	0.80 ± 0.03	0.99 ± 0.02	0.67 ± 0.77
<i>Undersampling techniques</i>				
ENN	0.680375	0.80 ± 0.03	0.99 ± 0.02	0.67 ± 0.77
NM-1	0.319625	0.48 ± 0.00	0.97 ± 0.00	0.32 ± 0.22
NM-2	0.84325	0.91 ± 0.02	0.99 ± 0.01	0.84 ± 0.28
RUS	0.716625	0.83 ± 0.04	0.99 ± 0.02	0.71 ± 0.74
T-L	0.680375	0.80 ± 0.03	0.99 ± 0.02	0.67 ± 0.77

Table 5 Comparison of SVM Classifier Performance using imbalanced and balanced data

	Accuracy	F-score	Precision	Recall
Imbalanced data	0.991625	0.99 ± 0.00	0.99 ± 0.00	1.00 ± 0.00
Balanced data	0.84325	0.91 ± 0.02	0.99 ± 0.01	0.84 ± 0.28

Table 5 demonstrates the comparison between the performance of the SVM classifier using imbalanced and balanced data. From Table 5, it is observed that imbalanced data fails to detect malicious activity. The Recall of SVM classifier using balanced data correctly predicts the non-malicious activity than imbalanced data. Precision and f-score are outperformed using balanced data, while accuracy remains satisfactory in the detection of malicious activity.

5 Conclusion and Future Enhancement

This proposed research paper implements different oversampling and undersampling strategies to combat the imbalanced class data with the classification prediction model. The CERT dataset includes the malicious insider threat is used, and SVM is applied for classification. The performance of the SVM classifier before sampling and after sampling is compared using various performance metrics. The performance of undersampling techniques outperformed the oversampling techniques to handle the imbalanced CERT dataset using SVM Classifier. NM-2 works better than other sampling techniques based on F-score and Recall. It eliminates majority class instances safely, resulting in improved performance than RUS, NM-1, ENN and T-L. In the near future, deep learning and other sampling techniques can be applied to classify the majority class instance of CERT data to improve performance.

Acknowledgements This work is supported by Centre for Cyber Intelligence (CCI), DST-CURIE-AI-Phase II Project, Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore, Tamilnadu, India - 641027.

References

1. Le D, Heywood Z (2020) Exploring anomalous behaviour detection and classification for insider threat identification. *Int J Netw Manage* 31(4):e2109
2. Devi D, Biswas SK, Purkayastha B (2020) A review on solution to class imbalance problem: undersampling approaches. In: 2020 international conference on computational performance evaluation (ComPE), pp 626–631
3. Gosain A, Sardana S (2017) Handling class imbalance problem using oversampling techniques: a review. In: 2017 international conference on advances in computing, communications and informatics (ICACCI), pp 79–85
4. Dittman DJ, Khoshgoftaar TM, Wald R, Napolitano A (2014) Comparison of data sampling approaches for imbalanced bioinformatics data. In: The twenty-seventh international FLAIRS conference, pp 268–271
5. Junsomboon N, Phienthrakul T (2017) Combining over-sampling and under-sampling techniques for imbalance dataset. In: Proceedings of the 9th international conference on machine learning and computing, pp 243–247
6. Hasanin T, Khoshgoftaar T (2018) The effects of random undersampling with simulated class imbalance for big data. In: 2018 IEEE international conference on information reuse and integration (IRI), pp 70–79
7. He H, Bai Y, Garcia EA, Li S (2008) ADASYN: adaptive synthetic sampling approach for imbalanced learning. In: 2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence), pp 1322–1328
8. Yap BW, Abd Rani K, Abd Rahman HA, Fong S, Khairudin Z, Abdullah NN (2014) An application of oversampling, undersampling, bagging and boosting in handling imbalanced datasets. In: Proceedings of the first international conference on advanced data and information engineering (DaEng-2013). Springer, Singapore, pp 13–22
9. Fujiwara K et al (2020) Over- and under-sampling approach for extremely imbalanced and small minority data problem in health record analysis. *Front Public Health* 8:178. <https://doi.org/10.3389/fpubh.2020.00178>
10. Bunkhumpornpat C, Subpaiboonkit S (2013) Safe level graph for synthetic minority over-sampling techniques. In: 2013 13th international symposium on communications and information technologies (ISCIT). IEEE, pp 570–575
11. Abdi L, Hashemi S (2015) To combat multi-class imbalanced problems by means of over-sampling techniques. *IEEE Trans Knowl Data Eng* 28(1):238–251
12. Elhassan T, Aljurf M (2016) Classification of imbalance data using Tomek link (T-link) combined with random under-sampling (RUS) as a data reduction method. *Global J Technol Optim S* 1:11
13. Glasser J, Lindauer B (2013) Bridging the gap: a pragmatic approach to generating insider threat data. In: 2013 IEEE security and privacy workshops, pp 98–104
14. Meng F, Lou F, Fu Y, Tian Z (2018) Deep learning based attribute classification insider threat detection for data security. In: 2018 IEEE third international conference on data science in cyberspace (DSC), pp 576–581
15. Pengfei J, Chunkai Z, Zhenyu H (2014) A new sampling approach for classification of imbalanced data sets with high density. In: 2014 international conference on big data and smart computing (BIGCOMP), pp 217–222
16. Guo H, Li Y, Shang J, Mingyun G, Yuanyue H, Bing G (2017) Learning from class-imbalanced data: review of methods and applications. *Expert Syst Appl* 73:220–239