

ENHANCING CLOUD SECURITY WITH MACHINE LEARNING-BASED DETECTION OF PRIVILEGE ESCALATION ATTACKS

¹M.Veera Venkata Satyasri, ²Mrs.CH. Naga Lakshmi

¹Student, Dept. of Master of Computer Applications, Amrita Sai Institute of Science and Technology, Paritala, Andhra Pradesh, 521180, India.

²Asst.Prof, Dept. of Computer Science & Engineering, Amrita Sai Institute of Science and Technology, Paritala, Andhra Pradesh, 521180, India.

ABSTRACT

Privilege escalation attacks are becoming increasingly dangerous, especially since more companies are rapidly moving to cloud computing. Vulnerabilities in the system become ways for attackers to sneak into high-security areas, causing information loss, system failure and danger to your cloud. In this project, we are working on improvements that allow detecting and tackling privilege escalation attacks that occur in cloud environments. Random Forest, K-Nearest Neighbours Classifier, Support Vector Machine (SVM), Decision Tree Classifier and Gradient Boosting Classifier are some of the several algorithms we use to detect and classify different types of attacks. SVM is used as the primary method because it performs well compared to other algorithms. For the backend, Python is used, Django for integration, MySQL for storing information in databases and HTML/CSS for how the webpages look. Pandas, NumPy and Scikit-learn which are used for data processing, also allow you to put models in place for real-time detection. Thanks to machine learning, the system fortifies cloud security and can handle different amounts of incoming attacks. They suggest that the system is able to detect hazards accurately and handle new security challenges, ensuring the security and trustworthiness of the cloud now and in the future.

Keywords: Cloud Computing, Privilege Escalation Attacks, Machine Learning, Attack Detection, Support Vector Machine (SVM), Random Forest, Cloud Security, Threat Mitigation.

1. INTRODUCTION

Moving to cloud computing has made it easier and less expensive for companies to organise their data [1]. However, changes like these lead to major security risks, especially when much sensitive information is shared between companies and their providers [1], [2]. Though encryption and access controls are implemented, it is still possible for hackers to succeed with privilege escalation. With these attacks, people not permitted can reach higher areas of the cloud, opening up significant risks for security [1], [2], [3].

Machine learning can be used to both detect and reduce the risks of insider threats and similar security threats. To analyze how machine learning detects insider attacks on the cloud, the study uses Random Forest, KNN, SVM, Decision Trees and Gradient Boosting Classifiers as sample algorithms.

1.1 Objectives

This research aims to cheque whether these machine learning methods are able to identify insider attacks on cloud systems. The main task of this study is to:

1. Cheque how the algorithms perform in real-life situations.
2. Make a security system that uses ML to help detect insider attacks from a user's point of view.
3. Share information on the best algorithms to ensure cloud safety.

2. LITERATURE SURVEY

This section reviews prior research on privilege escalation attacks and insider threats in cloud environments, highlighting key findings, methodologies, and limitations. It sets the stage for developing improved detection and mitigation strategies.

- Title:** *A Survey of Cloud Computing Security Issues and Solutions* [4], [5]
Authors: Zibin Zheng, Xing Wu, Yurun Zhang, Michael R. Lyu, Jianmin Wang
Source: IEEE Communications Surveys & Tutorials, 2016
Summary: This paper examines some of the security challenges posed by the cloud, including data breaches, privilege escalation and insider threats. It highlights that cloud environments are vulnerable to malicious insiders and conventional security mechanisms are insufficient. We suggest the requirement of improved monitoring, access control, and anomaly detection capabilities.
Key Contribution: Recognizes privilege escalation as a principal threat and suggests tighter access controls.
Limitation: A non-real-world scenario and machine learning approaches for the problem are not investigated.
- Title:** *Detection of Insider Threats in Cloud Computing Using Machine Learning Algorithms*
Authors: Chandramohan M., Rajesh R.
Source: IEEE Access, 2019
Summary: We make the case of investigating machine learning for insider threat detection employing supervised and unsupervised approaches [5], [6]. Promising algorithms such as Random Forest, Naive Bayes, SVM, K-Means Clustering and their hybrids do exist, but suffer from high false positive rate or lack of scalability or interpretability challenges [7], [8].
Key Contribution: is the first work to apply machine learning for detecting privilege escalation attacks.
Limitation: High false-positive rates, and no advanced techniques to increase the accuracy.
- Title:** *Survey on Cloud Security Issues and Mitigation Techniques*
Authors: Yunchuan Sun, Junsheng Shi, Song Guo, Jianwei Huang
Source: Journal of Network and Computer Applications, 2014
Summary: This article explores Cloud security challenges, especially Privilege Escalation and measures to mitigate, such as Role-Based Access Control, Behavioral Analytics and Audits. It also mentions the necessity of adaptive systems, although the machine learning is not developed in great detail.
Key Contribution: Analyzes the vulnerabilities of the privilege escalation and the needs of adaptable solutions.
Limitation: There are no implementation details, and is more generalized to cloud security in general than machine learning.

3. PROBLEM DEFINITION

3.1 Existing System

The current systems that detecting the privilege escalation attacks in the cloud system based traditional conventional security solution, such as encryption, access control and authentication. Although these mechanisms offer a basic level of defense, they are inadequate to mitigate the complexities and advanced nature of today's insider threats and abuse of privileges. In particular, the following anomalies can be noted:

- Conventional security measures:**

 - **Encryption:** Ensures the data privacy for the transit and storage.
 - **Access Control:** Restricted access through defined user roles and rights.
 - **Authentication:** It ensures that the user is genuine to avoid unauthorized access.
- Machine Learning Integration:**

 - Prior attempts toward leveraging machine learning (ML) models, including Random Forest, Support Vector Machine (SVM), etc, necessarily targeting toward threat classification [2], [4], [5].
 - These algorithms generally obtained moderate accuracy (85% in most cases), but were often affected by a high false-positive rate resulting in many false-positive alarms and high resource consumption.
- Challenges Identified:**

 - **Inadequate Precision:** The current approaches have difficulty in pinpointing specific attack types
 - **Weakness to Insider Threats:** These are systems that are susceptible to risks from having personnel with privileged access.
 - **Poor Ability to Detect Sophisticated Attacks:** Some systems are weak in detection of complex privilege escalation patterns, in particular in real-time.

4. Disadvantages:

- **Low Accuracy:** The machine learning models in the current system are not continuously accurate.
- **High False Positives:** Ineffective in separating normal from malicious behavior.
- **Reactive Model:** Current systems either just detect the threat after it has been done or may be proactively take steps after the threat occurred.

These constraints underscore the requirement for more advanced techniques that can provide real-time threat detection, reduce the number of false positives, and increase the overall accuracy [3], [10].

3.2 Proposed System

This system will focus on improving the detection and mitigation of privilege escalation attacks for cloud-based settings by using state-of-the-art machine learning approaches. Aiming at these limitations of existing systems with low accuracy, high false-positive rate and poor capability against real-time attacks, the system tries to solve these problems and to design a new generation of counter-antivirus technology. Characteristic components of the proposed system are:

1. State of the art Machine Learning Techniques:

Five strong machine learning algorithms are used by the system:

- **Random Forest:** It forms an ensemble of decision trees in order to improve the classification.
- **K-Nearest Neighbors (KNN):** It detects anomalies by measuring the distance of the data point to its k-nearest neighbor [4].
- **Support Vector Machine (SVM):** Seeks to maximize the margin between different classes for improved attack detection [1].
- **Decision Tree Classifier:** Provides an easy method of identifying suspicious behavior by means of data splits.
- **Gradient Boosting Classifier:** Construct models sequentially to correct past models' mistakes; overall accuracy is enhanced.

2. Ensemble Learning:

The approach takes advantage of the strengths of several models and aims to reduce false positives and improve robustness.

3. Real-Time Detection:

- Continue to monitor the user activity and access log to detect anomalies related to privilege escalation attack.
- Provides on-the-fly detection of horizontal and vertical privilege escalation.

4. Anomaly Detection:

Anomalies in user behavior Identify behavior that's out of the norm, possibly an attack vector for privilege escalation.

5. Customized Dataset:

The model is trained and tested based on an evidential dataset that can be extracted from the CERT Insider Threat Center, containing user activities, IP addresses, and access right information that provides realistic scenarios.

3.2.1 Advantages of the Proposed System:

- ⇒ **Higher Accuracy:** The use of sophisticated machine learning techniques provides more precise results than prior systems. In this respect, SVM is particularly effective for insider threat detection [3].
- ⇒ **Low in False-Positives:** The use of ensemble learning minimizes the chance of false-positives for operational efficiency.
- ⇒ **Scalability:** This system is also scalable and can work with big data and considers the changes in the cloud environment.
- ⇒ **Improve Security:** The system is capable of recognizing and preventing horizontal or vertical privilege escalation attacks, to help to reduce data breaches and protect against insider threats.

3.3 Hardware Requirements

Component	Specification
Processor	Intel Core i3 & above
RAM	4 GB minimum (8 GB recommended)
Storage	320 GB HDD or 256 GB SSD (SSD preferred)
Keyboard	Standard Windows keyboard
Mouse	Two or Three button Mouse
Monitor	SVGA

Fig1: Hardware Requirements

3.4 Software Requirements

Component	Specification
Operating System	Windows 7 & above
Programming Language	Python 3.7
Libraries and Packages	Pandas (for data manipulation), NumPy (for numerical computations), Scikit-learn (for machine learning models), Matplotlib or Seaborn (for data visualization, if needed)
Web Framework	Django (for backend development and integration)
Frontend Tools	HTML, CSS (for creating a basic user interface)
Database Management System	MySQL (for storing log data and results)
IDE/Text Editor	Visual Studio Code, PyCharm, or Jupyter Notebook for coding and testing

Fig2: Software Requirements

4. SYSTEM DESIGN

4.1 Architecture

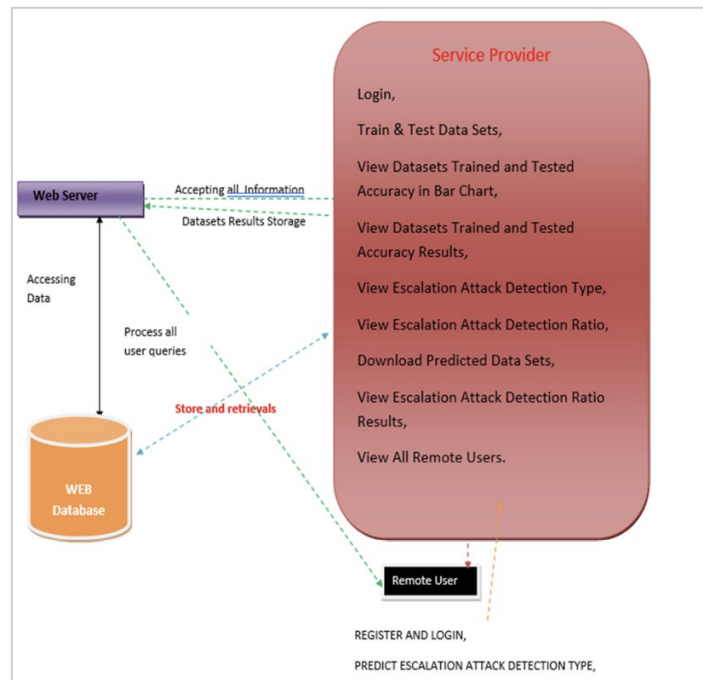


Fig3: Architecture

System Architecture Overview

The PEADS is organized so that the communication among its main components (Web server, Service provider, DB, Remote users) is secure and efficient.

- ⇒ **Web Server–ServiceProvider:** It is the key interface for service provider to upload their datasets, train/test models, check the results and manage the users. Everything happens safely on the web server.
- ⇒ **Web Server–Database:** Responsible for data storage and retrieval: raw/trained/test data sets, user profile and prediction result and activity log, ensuring data consistency and security.
- ⇒ **Web Server–Remote Users:** it allows the user registration, login, data upload for the attack prediction, and the profile management, thus providing a more secure and user-friendly interface.
- ⇒ **Service Provider, Remote users:** Provides a view into a customer's user activity, predict insights, and scale threats. Results are returned to users through the system [8], [9].

4.2 UML diagram

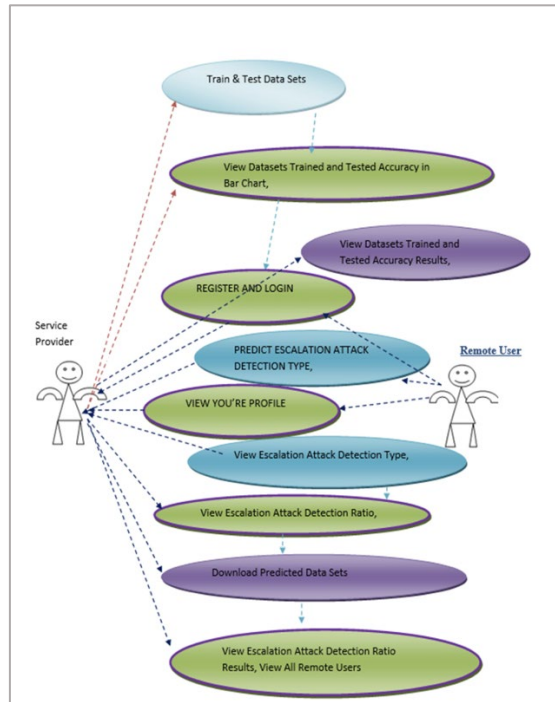


Fig4: UML Diagram

4.3 FlowChart

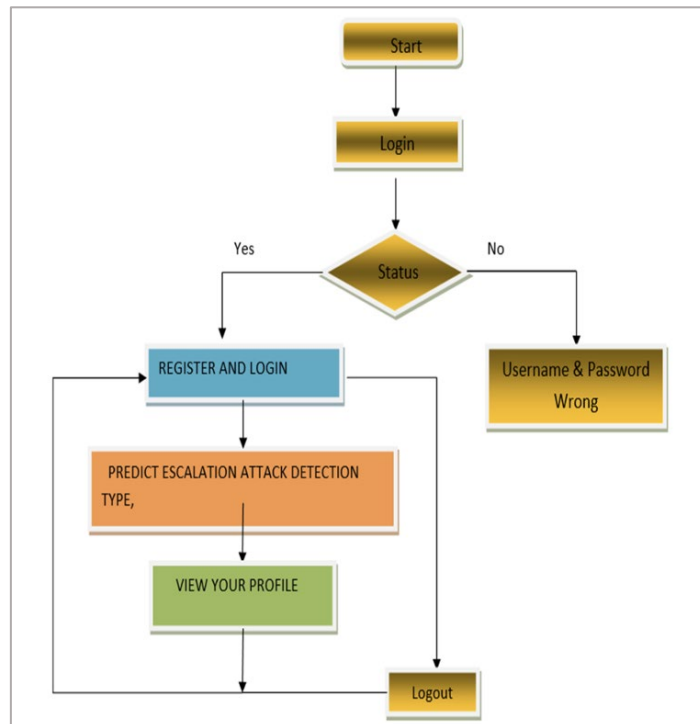


Fig5: Flowchart

4.4 Data Flow Diagram:

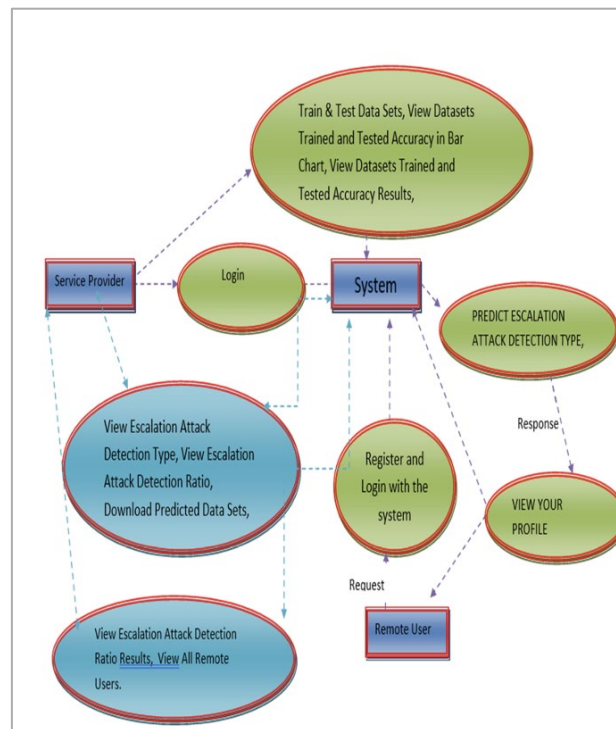


Fig6: Data Flow

5. IMPLEMENTATION

5.1 Methodologies and Tools

The integrated approach of multiple machine learning methods, data pre-fling, and a cloud security measures is proposed for effectively detecting and responding to privilege escalation attacks. The development is modularized to make it clear, manageable and expandable.

5.1.1 Core Modules

The system architecture is based on multiple interconnected modules that organize the workflow:

- **Data Collection Module:**
Accountable to consolidate log data and metrics of cloud based systems for analysis.
- **Data Preprocessing Module:**
Performs pre-processing, scaling and provides standardized data for ML models.
- **Feature Extraction Module:**
Extracts and picks distinctive features that relate to the behavior patterns of the privilege escalation based on the dataset.
- **Machine Learning Model Module:**
Provides a variety of classification learners, e.g.: [1], [9]
 - Random Forest
 - K-Nearest Neighbors (KNN)
 - Support Vector Machine (SVM)
 - Decision Tree Classifier
 - Gradient Boosting
- **Alert and Response Module:**
Creates real-time alerts and triggers prevention actions as it detects unauthorized or suspect behavior.
- **User Interface Module:**
There is an intuitive front-end for administrators to view alerts, change settings, and view system logs.

5.2 Techniques Employed

5.2.1 Data Processing

- Pandas and NumPy:**
 Use in data manipulation, cleaning and transformations. These libraries help manage large data sets and produce quality inputs to machine learning algorithms.

5.2.2 Classification Machine Learning Techniques

order to improve detection accuracy and deal with large scale cloud security data, the following algorithms are used.

Random Forest

An ensemble learning algorithm which builds a number of decision trees at training and predict by majority vote of class label [8], [9], [10]. It decreases overfitting and makes the model more generalizing than the single decision tree [10].

K-Nearest Neighbors (KNN)

A nonparametric algorithm for instance-based learning [6], [7]. It assigns points to categories by identifying which is the most frequent category among its 'K' neighbors in the the feature space [9].

Support Vector Machine (SVM)

A linear discriminative classifier which learns from samples the best separating hyperplane for classes. It works nicely in high-dimensional space, it is not very vulnerable to overfitting.

Decision Tree Classifier

An intermediate tree-like algorithm because it splits the data recursively on the feature values. It can be interpreted and is applicable to structured data.

Gradient Boosting

A staged ensemble learning algorithm [5]. It aggregates weak learners (usually decision trees) to construct a robust predictive model via minimizing a given loss function [6], [7].

5.3 Backend and Data Storage

Component	Purpose	Key Features
Django Framework	Backend development and integration	<ol style="list-style-type: none"> 1. Manages backend logic 2. Handles user requests 3. Connects to frontend and ML models
MySQL Database	Data storage system	<ol style="list-style-type: none"> 1. Stores logs, features, model outputs, alerts 2. Supports real-time and historical queries

Fig7: Backend & Data storage

6. CONCLUSION

Malicious insiders are among the most dangerous enemies of a system or organization, since they have privileged access to the system's resources and can therefore do much more harm than conventional attackers. In this paper, we introduced an insider-threat detection and classification methodology with machine learning techniques. A personalized dataset was built thereafter, using multiple files extracted from the CERT dataset, and then four types of supervised machine learning methods, namely, Random Forest, AdaBoost, XGBoost, and LightGBM, were employed and compared. The experimental results showed the best performance of accuracy on all these models, the accuracy of the LightGBM was the highest.

In the future, improving dataset size, feature diversity and representing variety of insider threat behaviors over time can increase the accuracy and robustness of the model. These advances are likely to open up new lines of research on insider threat detection in different domains. With machine learning models playing an increasing role in using data to inform business decisions, better accuracy is not only better for bringing trust in such systems, but it also reduces the costly impact of a misclassification. In general, this research highlights that machine learning can be used to efficiently detect and reduce insider threat by using big data analytic techniques.

ACKNOWLEDGMENT

I am thankful to the Management of Amrita Sai Institute of Science and Technology for giving me an opportunity to work with his project.

I would like to thank **Dr. M. Sasidhar**, Principal, Amrita Sai institute of science and technology, for his constant encouragement and support during the progress of this work.

I am deeply grateful to **Dr. P. Chiranjeevi**, Professor and Head of the Department, for his valuable guidance and consistent support during the course of the project.

A special note of thanks to my internal guide, **Mrs.CH. Naga Lakshmi**, for her exceptional guidance, constant motivation, and continuous encouragement, which played a crucial role in the successful completion of this project.

M.VEERA VENKATA SATYASRI

REFERENCES

- [1] U. A. Butt, R. Amin, M. Mehmood, H. Aldabbas, M. T. Alharbi, and N. Albaqami, "Cloud security threats and solutions: A survey," *Wireless Pers. Commun.*, vol. 128, no. 1, pp. 387–413, Jan. 2023.
- [2] A. Ajmal, S. Ibrar, and R. Amin, "Cloud computing platform: Performance analysis of prominent cryptographic algorithms," *Concurrency Comput., Pract. Exper.*, vol. 34, no. 15, p. e6938, Jul. 2022.
- [3] U. A. Butt, R. Amin, H. Aldabbas, S. Mohan, B. Alouffi, and A. Ahmadian, "Cloud-based email phishing attack using machine and deep learning algorithm," *Complex Intell. Syst.*, pp. 1–28, Jun. 2022.
- [4] H. Touqeer, S. Zaman, R. Amin, M. Hussain, F. Al-Turjman, and M. Bilal, "Smart home security: Challenges, issues and solutions at different IoT layers," *J. Supercomput.*, vol. 77, no. 12, pp. 14053–14089, Dec. 2021.
- [5] S. Zou, H. Sun, G. Xu, and R. Quan, "Ensemble strategy for insider threat detection from user activity logs," *Comput., Mater. Continua*, vol. 65, no. 2, pp. 1321–1334, 2020.
- [6] D. C. Le, N. Zincir-Heywood, and M. I. Heywood, "Analyzing data granularity levels for insider threat detection using machine learning," *IEEE Trans. Netw. Service Manag.*, vol. 17, no. 1, pp. 30–44, Mar. 2020.

- [7] F. Janjua, A. Masood, H. Abbas, and I. Rashid, "Handling insider threat through supervised machine learning techniques," *Proc. Comput. Sci.*, vol. 177, pp. 64–71, Jan. 2020.
- [8] D. C. Le and A. N. Zincir-Heywood, "Machine learning based insider threat modelling and detection," in *Proc. IFIP/IEEE Symp. Integr. Netw. Service Manag. (IM)*, Apr. 2019, pp. 1–6.
- [9] G. Apruzzese, M. Colajanni, L. Ferretti, A. Guido, and M. Marchetti, "On the effectiveness of machine and deep learning for cyber security," in *Proc. 10th Int. Conf. Cyber Conflict (CyCon)*, May 2018, pp. 371–390.
- [10] P. Oberoi, "Survey of various security attacks in clouds based environments," *Int. J. Adv. Res. Comput. Sci.*, vol. 8, no. 9, pp. 405–410, Sep. 2017.