Project Abstract

Cryptojacking Detection for Host-Based Systems in a Cloud Environment

Introduction

Cryptojacking is a type of cyber-attack where an attacker secretly uses a victim's computing resources to mine cryptocurrencies. This activity can significantly degrade system performance, increase operational costs, and shorten hardware lifespan. With cloud environments being increasingly targeted due to their high computational power and sometimes poorly configured servers, detecting cryptojacking becomes crucial to maintaining system integrity and efficiency.

Problem Statement

The project aims to detect cryptojacking attacks on host-based systems in cloud environments. Our project focuses on developing and evaluating machine learning models to detect cryptojacking attacks based on performance metrics from host-based systems in a cloud environment.

Objective

The primary objective of this project is to create an efficient, real-time detection system for cryptojacking attacks in cloud environments. By leveraging machine learning algorithms and performance metrics, we aim to develop a solution that can operate effectively within the dynamic and scalable nature of cloud infrastructures.

Detection Mechanisms:

- **Behavioural Analysis:** Monitor CPU, GPU, memory, and network usage for patterns typical of cryptocurrency mining.
- **Signature-based Detection:** Use known signatures of cryptojacking scripts and binaries.
- **Anomaly Detection:** Implement machine learning algorithms to detect deviations from normal resource usage patterns.

Dataset

We utilize the Cryptojacking Attack Timeseries Dataset from Kaggle, which includes:

- **Normal Dataset:** Performance metrics during normal operations.
- **Anormal Dataset:** Performance metrics during cryptojacking attacks.
- **Complete Dataset:** A combined dataset of normal operations and cryptojacking attacks

Methodology

1. **Data Preprocessing:**
   - A combined dataset of normal operations and cryptojacking attacks with an additional is_malicious column to label data points as normal (0) or malicious (1) ensuring accurate labelling.
   - Convert columns with mixed types to strings
   - Clean the data by handling missing values and normalizing performance metrics.
   - Split the dataset into 80% training and 20% testing sets for model evaluation.

2. **Feature Engineering:**
    - Extract and analyse key performance metrics that indicate cryptojacking activities.
    - Identify correlations and patterns in the data to improve model accuracy.
3. **Model Implementation:**
    - **Decision Tree:** An interpretable model that uses a tree structure to make decisions based on feature values.
    - **Random Forest:** An ensemble method that builds multiple decision trees and merges their results to enhance accuracy and reduce overfitting.
    - **K-Nearest Neighbors (KNN):** A non-parametric method that classifies data points based on the majority class of their nearest neighbors.
4. **Model Evaluation:**
    - Training each model using the training dataset.
    - Evaluate model performance using the testing dataset, focusing on metrics such as accuracy, precision, recall, F1 score, and computational overhead.
5. **Real-Time Detection:**
    - Develop methods for implementing real-time monitoring and detection with minimal computational impact.
    - Integrate the detection system into cloud infrastructures to ensure scalability and robustness.

Tools and Technologies Used

- **Python:** Primary programming language for data processing, model implementation, and evaluation.
- **Pandas and NumPy:** Libraries for data manipulation and numerical computations.
- **Scikit-Learn:** Machine learning library for implementing and evaluating models.
- **Jupyter Notebooks:** Development environment for interactive coding and data analysis.
- **Cloud Platforms (AWS, Azure, GCP):** For potential deployment and scalability of the detection system.

Expected Outcomes

- **Effective Detection Models:** Accurate machine learning models capable of identifying cryptojacking activities based on performance metrics.
- **Enhanced Security:** Improved security posture for cloud environments by mitigating unauthorized cryptocurrency mining.
- **Scalability and Efficiency:** A scalable and efficient detection system that can handle large volumes of data in real-time with minimal computational overhead.

Conclusion

This project aims to address the growing threat of cryptojacking in cloud environments by developing and implementing advanced machine learning models. By leveraging performance metrics and supervised learning techniques, we seek to enhance the security and operational efficiency of cloud infrastructures, ensuring robust protection against unauthorized resource exploitation for cryptocurrency mining.