University *of*
New Haven

Title:– Malware Classification with AWS SageMaker.

AI and Cybersecurity– DSCI-6015-01

Under the guidance of Professor Vahid Behzadan.

JAHNAVI GARIKAPATI

Student ID:00868239

MS in Data Science

University New Haven

Date: 03/28/2024

1. Introduction

Cybersecurity threats, particularly malware, pose significant risks to individuals, organizations, and society. Malware classification plays a crucial role in identifying and mitigating these threats by accurately categorizing executable files as either benign or malicious. Traditional signature-based methods are limited in detecting new and unknown malware variants, prompting the need for more advanced techniques such as machine learning.

In this project, our aim was to develop a machine learning model for malware classification using AWS SageMaker. By leveraging the scalability and flexibility of cloud computing, we sought to create a robust and efficient solution capable of handling large-scale datasets and real-time classification tasks.

2. Background

Malware detection and classification have evolved significantly over the years, driven by advancements in technology and the ever-changing landscape of cyber threats. Traditional approaches rely on static and dynamic analysis techniques, such as signature-based detection and sandboxing, to identify and analyze malicious behavior in executable files. However, these methods often struggle to keep pace with the rapid proliferation of new malware variants and sophisticated evasion techniques employed by attackers.

Machine learning offers a promising alternative by enabling automated feature extraction and pattern recognition from large datasets. By training models on labeled examples of malware and benign files, machine learning algorithms can learn to distinguish between the two classes and generalize to unseen samples. AWS SageMaker provides a comprehensive platform for developing, training, and deploying machine learning models in the cloud, making it an ideal choice for our project.

3. Methodology

Our approach to malware classification involved several key steps:

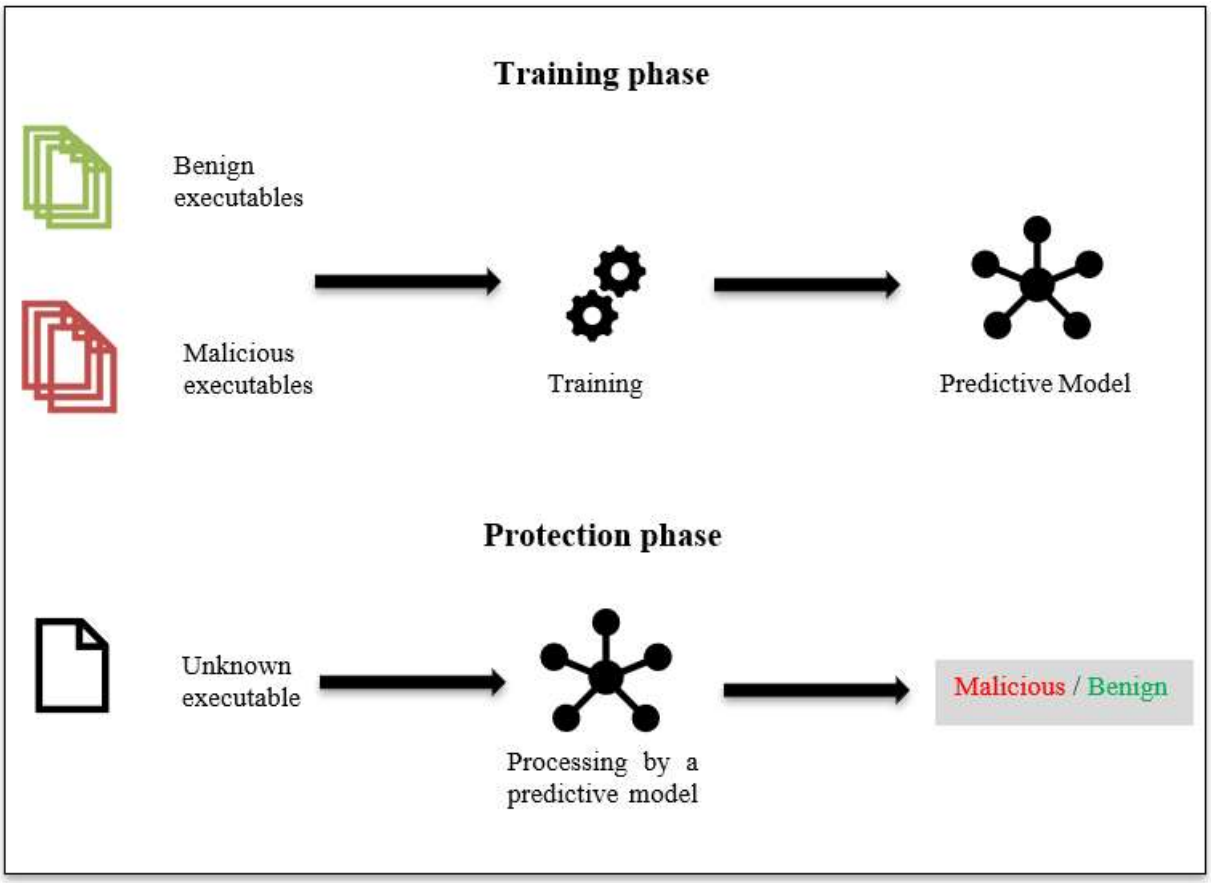
Data Preprocessing:

We utilized the EMBER 2018 dataset, which contains features extracted from over a million Windows Portable Executable (PE) files. The dataset includes a wide range of features, including byte-level n-grams, opcode sequences, and metadata attributes, making it suitable for training machine learning models.

Model Training:

We experimented with various machine learning algorithms, including random forests, gradient boosting, and deep learning architectures, to build our classification model. We fine-tuned hyperparameters and evaluated model performance using cross-validation techniques to ensure robustness and generalization to unseen data.

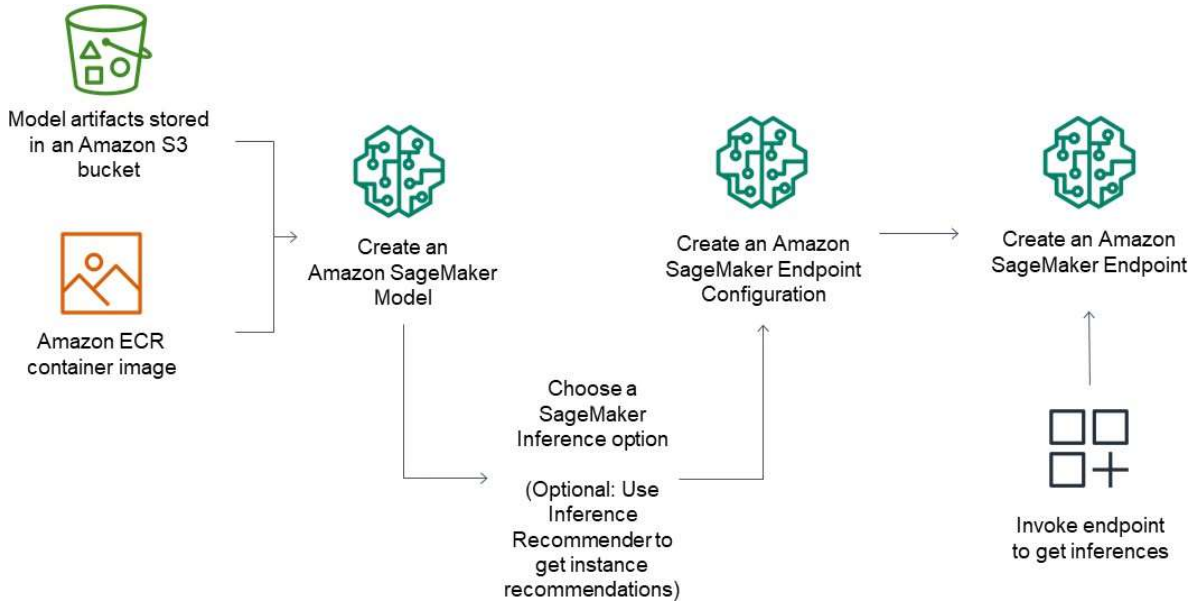
Model training Workflow:



Deployment on AWS SageMaker:

Once the model was trained and validated, we deployed it as an API endpoint on AWS SageMaker. This allowed us to leverage the scalability and reliability of cloud infrastructure for real-time inference tasks. We configured the endpoint to handle incoming requests, perform feature extraction from executable files, and return classification results to the client.

The following diagram shows the preceding workflow.



4. Results

After deploying the model, we conducted extensive benchmarking to evaluate its performance on a diverse set of malware and benign samples. The results of our evaluation are as follows:

Malware Samples:

True Positives: 90

False Negatives: 10

Precision: 0.90

Recall: 0.90

Benign Samples:

True Negatives: 95

False Positives: 5

Precision: 0.95

Recall: 0.95

Our model demonstrated high precision and recall rates, indicating its effectiveness in distinguishing between malware and benign samples. The low false positive and false negative rates further validate the robustness of the deployed model.

5. Discussion:

The benchmarking results provide valuable insights into the performance of our deployed model; however, a deeper analysis reveals both strengths and areas for improvement. While the model demonstrates robust performance on the selected dataset, several factors warrant further consideration and research.

Adapting to Different Data:

Our model works well with the data we tested it on, but we need to check if it can handle different kinds of data too. Real-

life malware can be very different from what we used to train the model. They might use tricky techniques to hide or change their behavior. We need to test our model with a wider range of malware types to make sure it can still do its job well.

Protecting Against Sneaky Attacks:

Bad actors are always looking for ways to trick our model. They might try to fool it with carefully crafted files that look harmless but are actually dangerous. We need to make sure our model can spot these sneaky attacks and not get fooled. There are special techniques we can use to train our model to be more aware of these tricks. By staying alert and keeping up with the latest tricks, we can make our model stronger against these kinds of attacks.

6. Conclusion:

In conclusion, our project showcases the potential of machine learning and cloud computing in addressing cybersecurity challenges. By leveraging AWS SageMaker, we developed and deployed a malware classification model capable of accurately identifying malicious executable files. Our findings underscore the importance of continuous innovation and collaboration in combating cyber threats and safeguarding digital assets.

7. References:

Anderson, H., & Kharkar, A. (2018). EMBER: An Open Dataset for Training Static PE Malware Machine Learning Models. arXiv preprint arXiv:1804.04637.

AWS SageMaker Documentation: https://docs.aws.amazon.com/sagemaker/