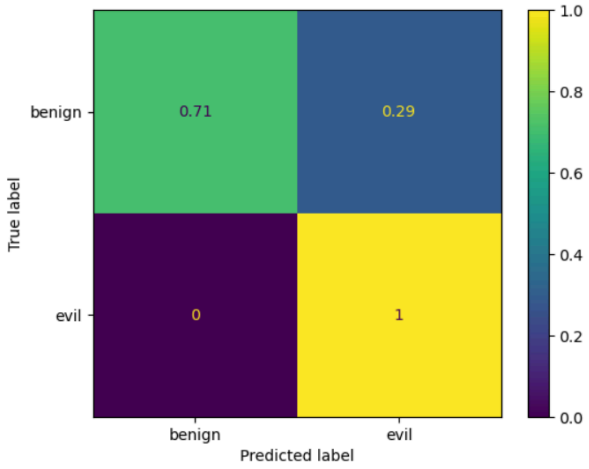
95767 Cybersecurity for Artificial Intelligence & Machine Learning

Assignment #1

Submitted by: Pragya Mittal
Andrew ID: pragyam

1) Interpretability - Describe what the visual shown in the last step (# Compute and plot performance metrics as a "confusion matrix") is representing. What meaning should a viewer draw from this and how is it determined?



Ans.

The confusion matrix is a table used to evaluate a model by making a visual comparison between its predicted labels and true labels. This shows how many of the instances were correctly classified and how many were misclassified.

It gives us an idea of the true positives where true label and predicted label are both evil (here 1), true negatives where true label and predicted label are both benign (here 0.71), false positives where the true label is benign and predicted label is evil (here 0.29), and false negatives where the true label is evil and the predicted label is benign (here 0).

In the confusion matrix, we can see that the rate of true positive is 1 which means that the model is classifying all the evil data correctly. However, the rate of false positives is 0.29 which is high meaning that the model is incorrectly classifying 29% of the data as malicious when it is not. This means that although the model is highly effective in classifying evil data, it will increase the workload of the SOC analyst who will further

inspect the traffic this model has classified as evil even when it is benign traffic, thus wasting precious man-hours and reducing their efficiency.

2) Explainability - Describe for a novice end-user how this model classifies evil and benign traffic in step "Fit an anomaly detecting isolation forest model to the engineered features". Consider what are the model inputs/features, how does the model process them, what is the output displaying?

Ans. This model uses the Isolation forest model to detect anomalies in the input data we give to it by identifying outliers in the input data. Here we are sending X as input to the isolation forest model which contains features from the dataset that have been preprocessed and converted into a usable format. These features are timestamp, processId, threadId, parentProcessId, userId, mountNamespace, hostname, processName, eventName, stackAddress, returnValue, and args. The isolation forest identifies anomalies or outliers in each of these features and if one data entry or one traffic has too many anomalies or outliers, the isolation forest classifies it as evil. After fitting the model, the isolation forest produces the anomaly scores and the predictions. The anomaly score is a number each data point gets that indicates how anomalous it is. The higher the score the more likely it is of being evil and vice versa. The model then classifies and labels each data point as evil or benign based on the anomaly score.

3) If an attacker knew you were doing anomaly detection such as in this Assignment for a network they planned to attack, how might they disguise their attack? Make your answer specific to the data available in the BETH dataset.

Ans. If an attacker knew that anomaly detection was being done on the basis of the features captured in the BETH dataset, they would try to evade detection by mimicking legitimate traffic. Following is what they can do for each feature to mimic legitimate traffic and evade anomaly detection:

- Timestamp: The attacker can send malicious traffic during office hours because the model looks for traffic at odd hours.
- processId, threadId, parentProcessId userId: The attacker can compromise a legitimate user account as part of a multi-stage attack and mimic legitimate process, thread, parent process, and user ID. This way the model will not be able to detect malicious traffic generated by the attacker.
- mountNamespace: The attacker can stick to the restricted mounted space that legitimate traffic is allowed to come from

The attacker can also choose the “low and slow” approach where they make minimal and infrequent calls to avoid detection

4) As a countermeasure, what step or steps might you add to your analysis to prevent the attacker from performing such disguise?

Ans. To detect timing attacks where the attacker sends malicious traffic during peak hours we could have enhanced timestamping analysis where we see not only the time at which a request was sent but also which device it came from. We can use this to set a baseline of the frequency, access times, and duration of requests generated and use that to detect anomalous or malicious requests.

In a similar vein, we can use User and Entity Behaviour analytics to understand the kind of traffic generated by a particular process, thread, parent process, or user ID and detect anomalous behavior even if it is coming from a legitimate source.

We can also look at process lineage to see if the parent ID of a process is what it should be in case the attacker is hiding their malicious activity by renaming their process ID to a legitimate one.

Enhancing monitoring of file and system interactions and integrity can help us detect if an attacker is generating malicious traffic from a legitimate mountspace. We can look for IOCs of a system and notice deviations like accessing an area outside of the typical mount space or sending an unusual number of system calls from the mountSpace.

Setting dynamic thresholds based on frequency analysis can help detect malicious activity if the attacker decides to go low and slow but this approach comes with caveats as it is essentially a cat-and-mouse game which is to say that the attacker can go lower and slower.

For detecting malicious traffic that is trying to evade detection using other methods we can use ensemble methods which combine multiple anomaly detection algorithms and implement a defense-in-depth strategy to reduce the likelihood of a false negative.

5) If you wanted to make this isolation forest model available to SOC analysts, how would you address the high rate of misclassifying benign records as malicious (i.e., false positives)? Your solution can include advice/guidance on how to use the model and/or technical workarounds.

Ans. The SOC analyst can be given access to the anomaly score along with the traffic classified as evil. The SOC analyst can then not spend a long time on traffic that has been flagged as evil and still has a relatively lower anomaly score.

We can include a feedback loop where the SOC analyst can let the model know that it falsely identified the benign traffic as evil which can be used to retrain the model readjust the threshold, and improve feature selection.

We can finetune and adjust the contamination parameter used in the isolation forest to make better predictions and match the actual percentage of malicious records and reduce false positives.

We can also cluster similar alerts together so the SOC analysts like alerts that came from the same source so they can decipher if the group of alerts is benign/has come from a legitimate source, they can discard those alerts and move on to analyzing other alerts and save time.

SOC analysts can also cross-reference the alerts raised by the isolation forest with those raised by other security systems like firewalls and IDSs. If the traffic flagged by the model is also flagged by the other security systems then it is probably actually evil and needs further inspection. SOC analysts can give a higher priority to alerts generated by multiple platforms and a lower priority to alerts generated just by the isolation forest.