ECE-GY 9163 ML for Security Project Report

Siyuan Shi: N18648680, ss13376 Haotian Yi: N18800809 hy1651

December 22, 2020

*Please note that we have implement two solution: a STRIP based method and a Fine-Pruning based method.

1 A Solution based on STRIP

1.1 Main idea

STRIP is a runtime detection method and its idea is based on the fact that trigger have strong effect to force result to be a fixed wrong class. For a test image, we superimpose it with random clean image for several times, if test image is poisoned, the result will be relatively static, otherwise the results will be chaos. And this is measured by entropy. Values of entropy of poison image is small and that of clean image is larger so we can compute a detection boundary.

1.2 Implementation

1.2.1 Main Functions & how they work

1. **superImpose**(overlay_img, origin_img, overlay_weight, back_weight):
Used to superimpose two images by weights, we use 0.5 and 0.9 as overlay_weight, back_weight so trigger won't be weaken.

2. **entropyCal**(background, clean_set, model, overlay_weight=0.5, back_weight=0.9):
Used to calculate mean entropy of 'background' image superimposed with randomly chosen image in 'clean_set'.

3. **getEntropyList**(x_test, x_valid, model, overlay_weight=0.5, back_weight=0.9):
Used to compute entropy lists of ‘x_test‘ superimposed with random images in ‘x_valid‘.

4. **computeThreshold**(entropy_benigh, frr=0.07):
Used to compute threshold (detection boundary) between clean and poison image. This is based on the assumption that entropy list is of normal distribution. We fit entropy list into a normal distribution and we assume False Reject Rate to be 7%, we set the threshold as the value at 7% of this normal distribution.

1.2.2 Detection Procedure

1. First we use **getEntropyList** (it will call **entropyCal**) to get a list of entropy for each test image. For each test image, we will superimpose it N times with randomly chosen clean image from validation set. In our implementation, N is set to 10 according to experience and referred material.
2. Then use **computeThreshold** to compute a detection boundary between entropy distribution of poison and clean image.
3. Use **entropyCal** (it will call **superImpose**) to calculate entropy for each test image, if the entropy is less than detection boundary, the test image is judged as backdoored image, otherwise it will be judged as clean, thus we can modify output to N+1 class.

1.3 Run code

Our code is initially conducted on Colab, but we have encapsulate the code into .py program. **Please run code according to readme.md.**

*Please note that STRIP is a runtime detection method, so we have test it with sunglasses-triggered model but not test with anonymous model, because we do not have test data with anonymous trigger.