Machine Learning Assisted Side Channel Analysis for Cryptographic Key Recovery

Team Members: Working alone

Introduction and Problem Summary

Side-channel analysis is a powerful approach to break cryptographic systems by exploiting information leaked through physical implementations. Modern encryption algorithms like AES are mathematically secure but their implementations on physical devices can sometimes leak sensitive information through power consumption, electromagnetic emissions. These leaks then can be measured and analyzed to determine the secret key in part or whole and from a penetration perspective this could easily be exploited via brute forcing or dictionary attacks.

Thus, this project will focus on applying machine learning techniques to understand the correlation between such traces and operations and hope to extract such keys in part or whole, to get a better understanding on how to prevent it.

Dataset Chosen: ASCAD/ATMEGA_AES_v1/ATM_AES_v1_fixed_key/Readme.md at master · ANSSI-FR/ASCAD

I have decided to use the ASCAD dataset, which is widely used and popular among such analyses. It contains side-channel measurements from AES-128 cryptographic implementations. Specifically, I'll focus on the ATMEGA_AES_v1 fixed-key version as provided in the GitHub repository.

It contains 60,000 electromagnetic traces in a HDF5 file containing raw electromagnetic measurements. Each trace contains 700 time samples representing electromagnetic emissions during the encryption operation.

This dataset's origin is the French National Cybersecurity Agency(ANSSI).

Solution Implementation:

I will first preprocess data and do feature selection and normalization. Then, I will implement and compare three major machine learning algorithms from our course syllabus:

1) Random Forrest
   This algorithm seems appropriate given the dataset's high dimensionality i.e almost 700 features. It will provide feature importance metrics to identify which portion of electromagnetic traces are most informative.

2) Support Vector Machines
   SVMs are highly useful for classification problem. It will help wot clear separation boundaries and I can employ different functions to capture the non-linear relationship in the data.

3) Neural Networks
   If time permits I will also implement Neural Network to learn the complex patterns. I will work on a simple forward architecture with maybe 1-2 hidden layers.

Evaluation: Each algorithm will be then evaluated using metrics like accuracy, precision, recall and F1Score.

A more sophisticated approach would be using CNN or ensemble learning methods, but for the sake of this project I will focus on the above topics and if time permits, explore other options.