CSE406: Computer Security Sessional
Side-Channel Attack

1. Assignment Overview

This assignment explores the concept of side-channel attacks in computer security, specifically focusing on website fingerprinting. By leveraging the timing and cache usage patterns, we will identify the websites a user visits without needing direct access to their screen or network traffic. The task involves several phases:

- ➢ **Timing Measurement**: The first step involves understanding the timing side-channel by collecting latency data based on memory access.
- ➢ **Trace Collection**: Implementing the Sweep Counting Attack, which measures cache access patterns to infer the website being visited in a separate tab.
- ➢ **Automation**: Using Selenium for automated data collection of traces from various websites, storing them in a database.
- ➢ **Machine Learning**: Applying machine learning techniques to classify websites based on the collected trace data.

The final goal is to implement and analyze the effectiveness of side-channel attacks for website fingerprinting, culminating in a report documenting the process and findings.

2. Design and Architecture

The system comprises:

- ➢ **Frontend**: Built with HTML, JavaScript, Alpine.js, and Pico CSS, it allows users to trigger trace collection, display latency data, and show heatmap visualizations based on the Sweep Counting Attack.
- ➢ **Backend**: Using Flask and SQLite, it handles API requests for collecting traces, storing data, and generating visualizations. PyTorch is used for training a machine learning model to classify websites based on side-channel data.
- ➢ **Data Collection**: Selenium automates the process of visiting websites, collecting side-channel data (timing and cache usage), and storing it in a database.
- ➢ **Side-Channel Attack**: The Sweep Counting Attack measures cache access patterns to infer the website being visited.
- ➢ **Machine Learning**: The collected data is processed and used to train a model that classifies websites based on side-channel traces.
- ➢ **Workflow**: The user triggers data collection, which is processed and visualized, with results documented in a final report.

3. Data Collection

Traces were collected from three websites:

1. https://cse.buet.ac.bd/moodle/
2. https://google.com
3. https://prothomalo.com

A total of 3,750 traces were collected, with 1,250 traces gathered from each of the three websites: https://cse.buet.ac.bd/moodle/, https://google.com, and https://prothomalo.com. These traces represent side-channel data collected through the Sweep Counting Attack, providing valuable insights into the memory and cache access patterns associated with each website. The dataset has been used to train and evaluate a machine learning model for website classification based on these patterns.

4. Experimentation Results

To evaluate the effectiveness of cache-based side-channel website fingerprinting,two convolutional neural network architectures **SimpleCNN** and **ComplexCNN** were trained and tested using trace data (3,750 traces)

Results from Trace Data :

Simple CNN :

Accuracy: 95.07%

Best Performing Site: prothomalo.com with 0.99 f1-score

```
Training model: SimpleCNN
Best test accuracy for SimpleCNN: 0.9507

Classification Report:
                              precision    recall  f1-score   support

https://cse.buet.ac.bd/moodle/      0.92      0.96      0.94       250
           https://google.com       0.95      0.90      0.92       250
       https://prothomalo.com       0.99      0.99      0.99       250

                     accuracy                           0.95       750
                    macro avg       0.95      0.95      0.95       750
                 weighted avg       0.95      0.95      0.95       750
```

Complex CNN:
Accuracy: 98.13%
Best Performing Site: prothomalo.com with 1.00 f1-score

```
Training model: ComplexCNN
Best test accuracy for ComplexCNN: 0.9813

Classification Report:
                              precision    recall  f1-score   support

https://cse.buet.ac.bd/moodle/      0.98      0.97      0.97       250
           https://google.com       0.97      0.98      0.97       250
       https://prothomalo.com       1.00      1.00      1.00       250

                     accuracy                           0.98       750
                    macro avg       0.98      0.98      0.98       750
                 weighted avg       0.98      0.98      0.98       750
```

The results show that a small, well-collected dataset can achieve high accuracy. The Complex CNN model excelled due to its deeper architecture and ability to capture detailed patterns.

Model comparison :

```
Model Comparison:
SimpleCNN: Best Test Accuracy = 0.9507
ComplexCNN: Best Test Accuracy = 0.9813
```

5.Discussion

The experiment showed that even a small, well-collected dataset can yield high accuracy. SimpleCNN achieved 95.07% accuracy, while ComplexCNN reached 98.13%, with the latter excelling due to its deeper architecture that captured finer patterns. The 3,750 traces from three websites were sufficient for training a reliable model, demonstrating the importance of model complexity in side-channel attacks.

.

Best model link :
https://drive.google.com/file/d/1GYo0MhHn6at7lXPI-PMJvPwh03o7nPzu/view?usp=drive_link

Dataset link:
https://drive.google.com/file/d/1myWpmv3gMX5Rdt62bOYswf0EitlVPJ-G/view?usp=sharing