**IMPLEMENTATION OF SYMMETRIC ALGORITHM MODIFICATION SYSTEM TO RESIST POWER BASED SIDE CHANNEL ATTACKS**

**Software Requirements Specification (SRS)**

Project Id: 17-044

IT14100376 - Hangawaththa N.H.A.D.A

B.Sc. Special (Honors) Degree in Information Technology

Department of Information Technology

Sri Lanka Institute of Information Technology

**IMPLEMENTATION OF SYMMETRIC ALGORITHM MODIFICATION SYSTEM TO RESIST POWER BASED SIDE CHANNEL ATTACKS**

Project Id: 17-044

Project Proposal Report

Pathirana K.P.A.P Lankarathne L.R.M.O Hangawaththa N.H.A.D.A

SUPERVISOR

…………………………………………………..

Mr. Kavinga Yapa Abeywardena

CO-SUPERVISOR

…………………………………………………..

Mr. Nuwan Kuruwitaarachchi

B.Sc. Special (Honors) Degree in Information Technology

Department of Information Technology

Sri Lanka Institute of Information Technology

**DECLARATION**

I declare that this is my own work and this SRS does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any other university or Institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

Hangawaththa N.H.A.D.A ……………………… Signature

**1 Introduction**

**1.1 Purpose**

The purpose of this document is to specify the approach of developing analyzing unit which is used in symmetric algorithm modification system. This software requirements specification document provides all the functional and non-functional requirements for analyzing unit. This document mainly focused on data classification, data preparation, data transformation to suitable machine learning methods, detecting any patterns in given datasets and performance utilization in in analysis. This document can be used as a guideline for developing analyzing unit for software Developers and Information Security Professionals

**1.2 Scope**

The scope of this document is limited to the development of the analyzing unit in symmetric algorithm modification system. In this document it covers the design criteria of data classification, data preparation, transforming to suitable ML techniques, identifying power consumption patterns and performance utilization.

The main reason to implement this unit is to analyze symmetric algorithms using ML techniques in order to identify any power consumption pattern from the dataset received from the power consumption measuring hardware device. Once the dataset is received from the hardware device, measurements are first classified by selecting the subset of all available data in a linear manner procedure. Classified data is prepressed in order to have well organized data set. This component will transform data set to the suitable ML technique and identify whether there is any pattern recedes in that particular data set.

After identification of the power consumption pattern, next step is to trace down the exact location in source code which suitable to add randomness to code. This document describes relevant performance utilization methods in order to increase the accuracy in patterns identification using ML techniques.

**1.3 Overview**

Side Channel attacks are an easy way to launch a powerful attack against cryptographic modules by examining its physical specificities without analyzing cipher text or plain text. Sensitive information of a cryptographic module can be easily tracked by evaluating the side channel information such as power consumption, heat, and electromagnetic emissions that outputs from the cryptographic device. Side channel attacks are getting much more popular since it is easy to mount an attack in a short time with only a few hundred dollars’ worth of devices and it is time saving comparing to other methods of breaking encryption also deep knowledge about algorithms isn’t required to perform an attack.

Symmetric algorithm modification system is a successful mitigation plan for this side channel attacks. This will identify patterns by measuring power consumption values and add randomness to the algorithm by adding random code segments to the source code.

This document contains two other sections. The second section will give an overall idea of analyzing unit and in-depth details which planned to develop under this module. It consists all the requirements that required to come up with analyzing unit and it describes how this module interact with other modules as well.

The third chapter gives a holistic view of the development process of the relevant specific sections in implementing analyzing unit. Also it gives details about the technologies and techniques that are planned to use in this module.

**1.3 Definitions, Acronyms, and Abbreviations**

|  |  |
| --- | --- |
| DD | Design Document |
| ML | Machine Learning |
| SCA | Side Channel Attacks |
|  |  |
|  |  |
|  |  |

Table 1: Definitions, Acronyms and Abbreviations

**1.4 References (place this at the end of the document)**

**2 Overall Descriptions**

In this section will give an overall idea about the whole analyzing unit and a clear view about the specifications that are discussed in this document. This will show how the each step is going to involve to come up with the main functionality that is covering from the analyzing unit.

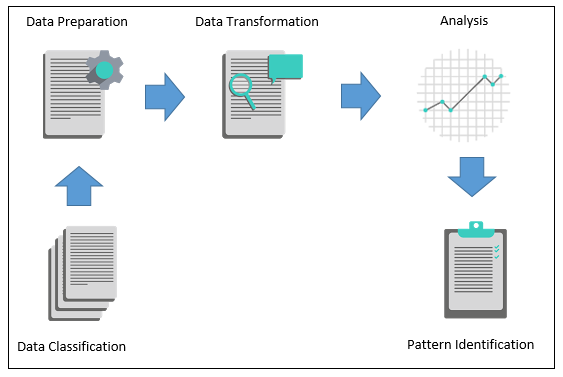
Figure 01 shows the overall picture of the analyzing unit in the Symmetric algorithm modification system.

Figure 1: High Level Diagram for analyzing unit

The analyzing unit is a standalone python application which identify power consumption patterns that can be used to derive an internal key from the power consumption measurements. After the confirmation about a pattern, it identifies the exact location in the source code which causes to generate that identified pattern.

Going into deep in this component, first step is to select the subset of all available data that can be used to find out most accurate results. By identifying extent of the data, most required data as well as less important data, it easy to select the subsets of datasets. After Selecting data, data should be preprocessed using different techniques. Formatting, cleaning and sampling are the techniques that will use to prepare the datasets.

After data prepressed, data is transformed to suitable machine learning techniques in order to analyze whether those preprocessed dataset contains any patterns which can be caused to launch aside channel attack. Supervised and unsupervised machine learning techniques will be used in analyzing datasets.

After finding a particular pattern by examining couple of datasets repetitively using supervised and unsupervised machine learning algorithms, this module will identify the exact location in the source code which is much more effective to add random source code in order to mitigate that identified pattern. After Identified the source code location modification unit add random source code in order to mitigate the pattern without altering to the final output of the algorithm.

In this document, only the functionalities relates to analyzing unit is described. Implementation of power measuring unit and modification unit are separate two modules which will be not discussed in this document as they are two separated modules.

**2.1 Product perspective**

There are other products in the market which are related to power based side channel attacks like **SASEBO (Side channel Attack Standard Evaluation Board), CHIP WHISPERER and SAKURA-G.** All of these devices are capable of performing a side channel attack by identifying patterns from the target devices and can derive a key or couple of partial keys. These devices have a very advanced design and complex design and they are not solely based on side channel attacks and can perform other kinds of attacks like Fault Injection Attacks (FIA), Physical Unclonable Function (PUF) and dynamic reconfigurations etc… and these devices cost around $300 – $400.

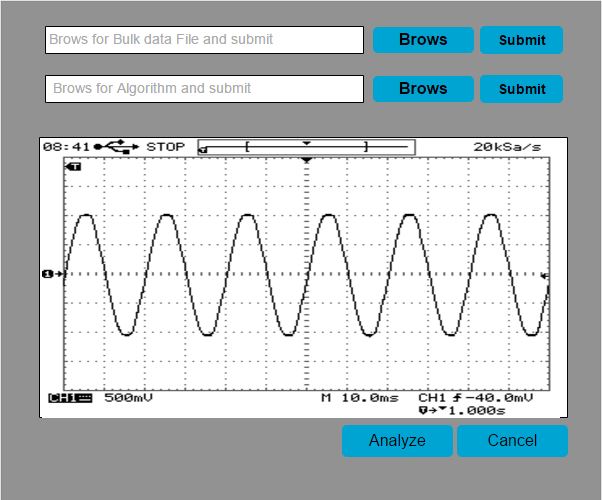
The main difference between our system and the system that mentioned above is, though these system in the market currently can implement a side channel attack they are not capable of providing a method prevent side channel attacks by finding an algorithms vulnerabilities and fixing them. And another difference is that our device is the design is minimalistic and simple with specific functionalities that is only relating to power consumption analysis which enables us to reduce the cost of the device. Any person with basic computer literacy can mount the device and run the necessary programs to modify the algorithm to secure it from power based SCA since all the tasks in the system are automated.

2.1.1 System interfaces

This product is going to be developed for Microsoft Windows environment. The main interface of Symmetric algorithm modification system will be developed by Java and python.

2.1.2 User interfaces

As all the functionalities in analyzing unit is mainly done in back-end as a service. All the services are mainly done through API calls in python. So there are no separate user interfaces that helps to interact with end user for each functionality in analyzing unit. But as per the figure 2.0, user can initiate analyze process by clicking a button.



2.1.3 Hardware interfaces

No special hardware interfaces are needed to operate analyzing unit. But all the datasets which are received to analyzing unit is from implemented hardware device which measure power consumption values.

2.1.4 Software interfaces

Several open source software tools will be used to develop analyzing unit. Python 3.6 will be used as the developing language and Java will used in creating user interfaces. Machine learning python APIs will be used in developing analyzing unit.

2.1.5 Communication interfaces

Analyzing unit is running on the PC and power consumption measuring unit is directly connected to the PC and retrieve those datasets via USB. Other than that, there is no any separate communication interfaces in analyzing unit.

2.1.6 Memory constraints

The endpoint system need to have a minimum 1024MB RAM for running analyzing unit without drawbacks.

2.1.7 Operations

This component does not require interacting with users to perform its every operations. User can initiate analyze process once the power consumption measurement s are received to the analyzing unit.

2.1.8 Site adaptation requirements

*Comment: Requirements for execution on a particular installation*

**2.2 Product functions**

Analyzing unit consists with several functions,

1. Data classification
2. Data preparation
3. Data transformation and analysis
4. Performance utilization

**2.3 User characteristics**

A user can be a novice or a computer related personal, whose intention is to mitigate the side channel attack possibility in their symmetric algorithms that used to encrypt sensitive information. Symmetric algorithm modification system is focused on simplicity that any user can use this tool. The user does not need to have a sound knowledge about security or IT.

**2.4 Constraints**

*Comment: All conditions that may limit developer's options. These can originate from many sources.*

**2.5 Assumptions and dependencies**

Currently tool is developing for windows platform. Future development will cover up Linux platform.

**2.6 Apportioning of requirements**

*Comment: Order in which requirements are to be implemented.*

**3 Specific requirements**

## 3.1 External interface requirements

### 3.1.1 User interfaces

Comment: Description of user interface in section 2.1.2 showed only sketches of user interfaces in order to provide product perspective. It lacks details and should not be regarded as the last word. If user interfaces are not completely specified later in this document, then all details should be given in this section.

### 3.1.2 Hardware interfaces

As mentioned in the 2.1.3 there is no special hardware requirements.

### 3.1.3 Software interfaces

As mentioned in 2.1.4 Python 3.6 will be used as the developing language and PyGUI will be used to develop the graphical interface of the tool.

### 3.1.4 Communication interfaces

As mentioned in 2.1.5 other components in the R-Killer tool has several communication interfaces with the public network. But DPE does not have any communication interfaces.

## 3.2 Architectural Design

### 3.2.1 High level Architectural Design

### 3.2.2 Software requirements with justification

### 3.2.3 Risk Mitigation Plan with alternative solution identification

### 3.2.4 Cost Benefit Analysis for the proposed solution

## 3.3 Performance requirements

Comment: Performance requirements include required speeds and/or time to complete. Unless documented in a different section of the SRS, they may also include memory usage (RAM and/or disk) noted either statically or dynamically (i.e., memory required at runtime).

## 3.4 Design constraints

Comment: Restrictions on design. If there is no material in this section, designers are free to create any (good) design that satisfies the requirements.

## 3.5 Software system attributes

### 3.5.1 Reliability

### 3.5.2 Availability

### 3.5.3 Security

### 3.5.4 Maintainability

Comment: Lists of all functions/classes that are expected to change soon or to change frequently.

## 3.6 Other requirements

# **4 Supporting information**

## 4.1 Appendices

Comment: This may include supporting or background information that can help the readers of SRS, description of the problem, special packaging instruction for the code and the media, results of user surveys, sample I/O formats, ...