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

Software Requirements Specification (SRS)

Project Id: 17-044

IT14116216 – L.R.M.O Lankarathne

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.

IT14116216 – Lankarathne L.R.M.O ……

Signature

SUPERVISOR

……..
Mr. Kavinga Yapa Abeywardena

CO-SUPERVISOR

……..
Mr. Nuwan Kuruwitaarachchi

Table Of Contents

1 Introduction

1.1 Purpose

This software Requirements Specification (SRS) document deliver the all functional and the non-functional requirements for the IMPLEMENTATION OF SYMMETRIC ALGORITHM MODIFICATION SYSTEM TO RESIST POWER BASED SIDE CHANNEL ATTACKS Project's Automate Code embedding mechanism using machine learning phase requirements. All Parts of this phase are planned and developed an automated mechanism to add suitable modifications to the analyzed algorithm.

1.2 Scope

This document covers the Automated Code Embedding phase of the IMPLEMENTATION OF SYMMETRIC ALGORITHM MODIFICATION SYSTEM TO RESIST POWER BASED SIDE CHANNEL ATTACKS Project. The Objectives of this phase is to develop an automated mechanism to add suitable modifications to the analyzed algorithm. This is more important as all the modifications should be appended to the algorithm in code level without creating and impact of the output of the algorithm and its security features.

With the Automate code embedding phase in above mentioned project, it is needs to identify the necessary facilities, automation techniques, Choose appropriate ML algorithms to trains the module, implement the automate code generation and Enhancing utilization and performance of the ML algorithm. By adding all these functionalities, this project will give the securing encrypted data of a company by modifying the algorithm uses for encryption to resist power based side-channel-attacks (SCA).

1.3 Definitions, Acronyms and Abbreviations

ML	Machine Learning
SRS	Software Requirement Specification
SCA	Side Channel Attacks
ACE	Automate Code Embedding

PMU	Power Measurement Unit
AU	Analyzing Unit
MU	Modification Unit

1.4 Overview

As side channel attacks are the attacks based on side channel information which are the information that can be retrieved from the encryption device that is neither plaint text nor the cipher text resulting from the encryption process. These side channel information can be collected from heat, sound, electromagnetic emission, power consumption etc... By use these information to analyze an encryption algorithm to get information like encryption key and other sensitive information related to the protected information.

Mainly this project based on power consumption out of the other side channel information. Because among other side channel information, power consumption analysis is one of the most effective and powerful method. And it is the most frequently used method to collect side-channel-information by attacker and intruders because power consumption analysis is also considered as one of the most reliable side-channel-information sources and if the cost is considered and compared with other methods of information gathering (electromagnetic, noise etc…) power consumption is the most cost effective.

Through this product, securing the encrypted data of a company by modifying the algorithm uses for encryption to resist power based side-channel-attacks. For that we identify the possible vulnerabilities for a given algorithm using power analysis next implement the necessary counter measures by modifying the algorithm.

This document contains three chapters including this. Second section provides an overall idea of the ACE and in-depth details of the ACE planned. It consist of the system functionality, requirements, system interaction with each other phases and interact with the user.

2 Overall Descriptions

In this section will give an overall idea about the whole system. The product will be explained in its context to show how the system interact with other systems and introduce the basic functionality of it.

The side channel attacks is not a new concept for the cryptography. Currently one of the most trending cryptanalysis method that been used is side-channel-attacks. An attacker always does not need to find the keys by complex mathematical calculations in order to break encryption because the cryptographic algorithms are not purely mathematical objects they also need to be implemented. In SCA these information also take in to the account. Cryptographic algorithms are implemented on software but also on physical devices, these devices will interact with the environment and the changes in the environment around the device can be closely monitored and gather cryptanalysis information like patterns in noise, electromagnetic emissions, power consumption and these are called side-channel-information. These information can be used to break cryptographic algorithms and they are called side-channel-attacks.

Following Figure 1, shows the overall picture of this project. It represent the major components and functionalities of this project and it is available as standalone application and it based on java and python language. This will produce as a solution to mitigate the power based side channel attack which means mitigate the most common side channel attack.

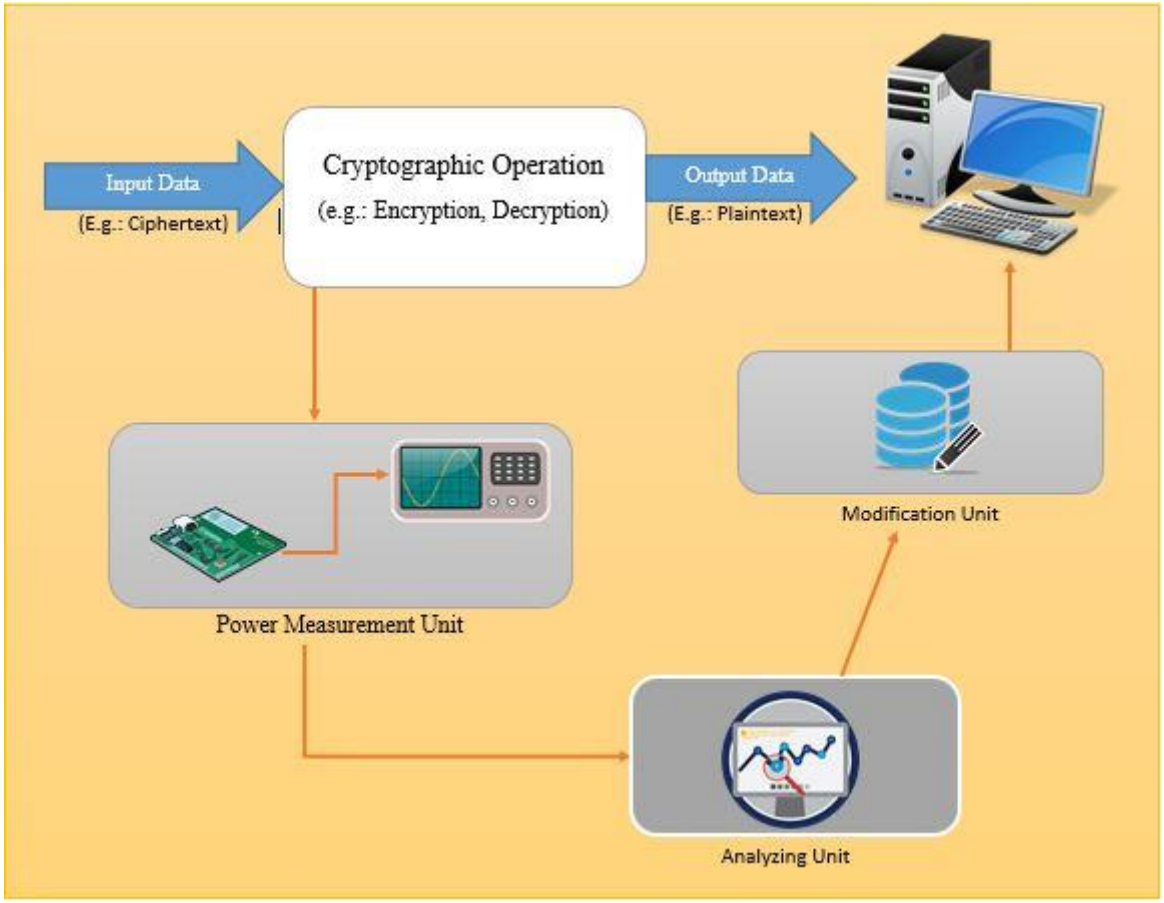

Figure-1: Overall Picture of the Project

This overall diagram shows the 3 phases of this project. It contains Device implementation for extracting power consumption measurements it called as Power Measurement Unit (PMU), Data classification/analysis and module training using machine learning it called as Analyzing Unit (AU) and Automate Code Embedding mechanism using machine learning it called as Modification Unit (MU).

Through the MU developed an automated mechanism to add suitable modifications to the analyzed algorithm according to the suggested countermeasures from the trained data analysis module. All these modifications add to the algorithm without creating any impact to the output of the algorithm and its security features.

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 Automate code embedding unit is mainly done in back-end as a service. All the services are mainly done through Methods calls in python and java. So there are no separate user interfaces that helps to interact with end user for each functionality in Automate code embedding unit. Following Figure-2 shows the interface that is appear after complete the modification process and fix the vulnerable points in the algorithm.

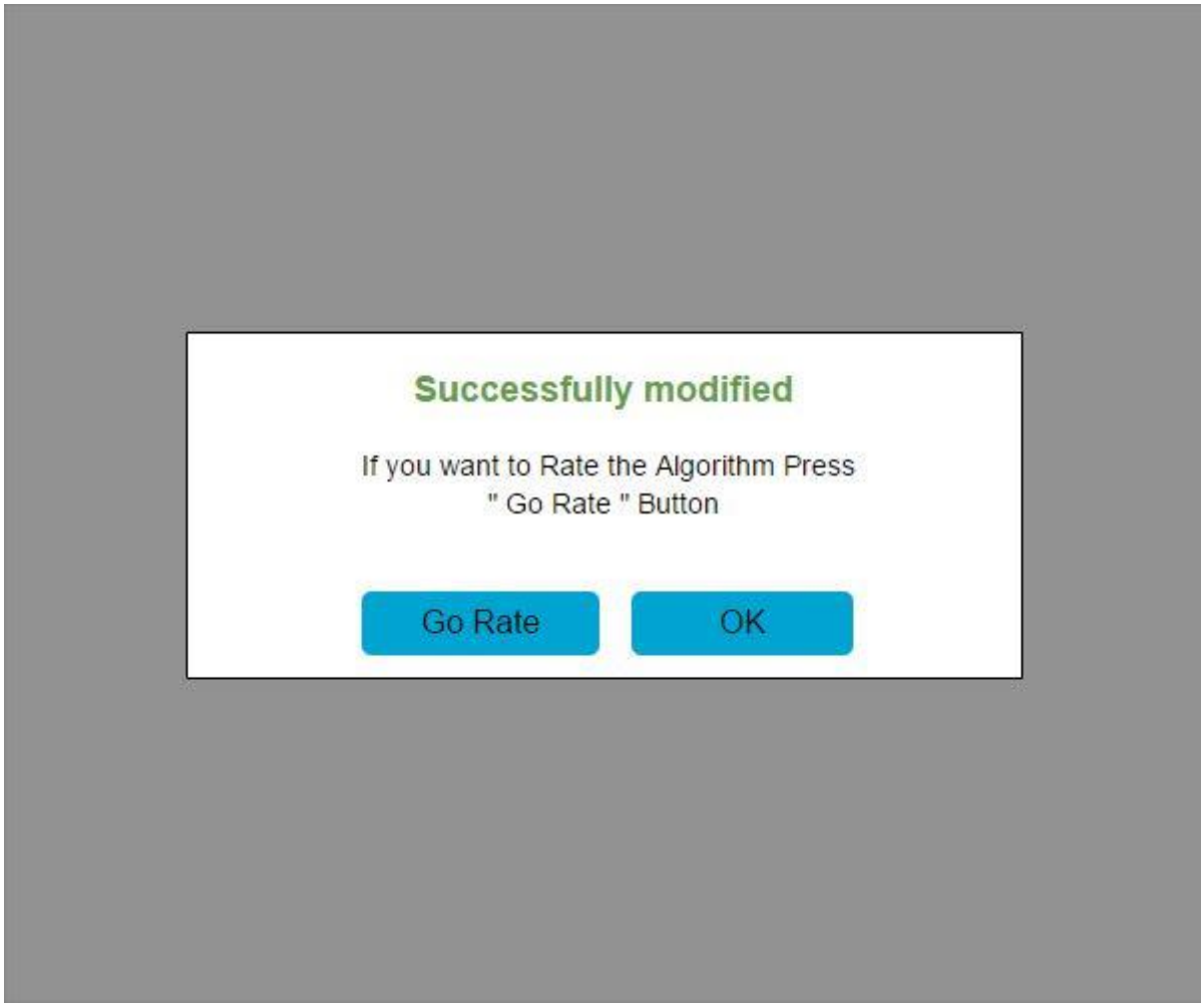

Figure-2

According to the above interface there's a button called "Go Rate". This button available for go to the algorithm rating section. This algorithm rating section implement as a by product of this project.

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 ACE unit. Python 3.6 will be used as the developing language and Java will used in creating user interfaces as well as backend developing. Machine learning python APIs will be used in developing ACE unit.

2.1.5 Communication interfaces

All phases of this project are running on the PC and power consumption measuring unit is directly connected to the PC and retrieve those datasets via USB and analyzing Unit passed the analyzed dataset to ACE. Other than that, there is no any separate communication interfaces in ACE unit.

2.1.6 Memory Constraints

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

2.1.7 Operations

This component does not require interacting with users to perform its specific operations. However, in another component of this project user requires giving file that bulk data include and also need to brows the algorithm path and select the algorithm.

2.1.8 Site Adaptation Requirements

To perform this component a windows environment is needed and internet connection is not necessary to perform this component.

2.2 Product Functions

This ACE unit contains with several functions.

- Identifying the suitable facilities to come up with ML techniques.
- Learning necessary techniques for automation.
- Choosing appropriate ML algorithms to train the module.
- Implementing automate code generation.

- Enhancing utilization and performance of the ML algorithms.

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

This system handles the users or company confidential algorithms. So under any circumstance the system should not creating any impact to the output of the algorithm and only break the power consumption pattern through this process and also should not share those critical information with the external parties.

2.5 Assumptions and Dependencies

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

2.6 Apportioning of requirements

This release of the symmetric algorithm modification system is only focused on the power consumption measurements in order to identify any particular patterns. So with future releases, we hope to come with different side channel information ratios like heat, sound and electromagnetic waves to have a multiple side channel measurements in order to increase the accuracy of the results.

3 Specific Requirements

3.1 External interface requirements

3.1.1 User interfaces

As this ACE unit is developed with minimal user interaction. So there is only one user interface which uses to show the process outcome status and button for the net process.

3.1.2 Hardware interfaces

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

3.1.3 Software interfaces

As mentioned in 2.1.4 python will be used as the backed developing language and java will be used to develop the graphical interface of the system and also to develop the backend.

3.1.4 Communication interfaces

As mentioned in 2.1.5 other components in this project has only USB communication interfaces to directly connect to the PC. So there is no communication interfaces regarding to the ACE.

3.2 Architectural Design

3.2.1 Software requirements with justification

To develop this system will be use Netbeans as open source tool and also we use free and open source python libraries as ML techniques. So no need to pay for get a licenses. So it reduces the developing cost of this project.

3.2.2 Cost Benefit analysis for the proposed solution

At the initial stage this product will produce as free and open source. And we going to promote this through social media and then pricing will be added to the upcoming services afterwards.

3.3 Performance requirements

This product is developed to run on Microsoft Windows platform. This will require a PC with some computational power and memory.

When the ACE unit workout it will consume processor and memory. For that PC which contains the application should at least match the below criteria.

- Processor: 1 GHz
- RAM: 1GB

3.4 Design constraints

This application need to run in administrative privileges in order to obtrude and identified the process. Because these algorithms are confidential information to the company if these algorithm details are handover to the some external parties like competitors of the company it can be serious issue to the company.

3.5 Software System attributes

3.5.1 Reliability

The Modification and ACE unit process must be reliable and should reduce create any impact to the output of the algorithm. Otherwise it will be useless for users when it made some impact to the output of the algorithm. To give reliable output, need correct power consumption reading and suitable data analysis is must.

3.5.2 Availability

The ACE unit must have not present the run time bugs after implementation of the project and it should be perform without giving any Errors.

3.5.3 Security

We need to properly validate the interface text fields to protect the system and the confidentiality of the algorithm. This system developed as a standalone application, according to that system not need any internet connection to perform the task. So that system will protect from the attacks that coming through the internet.

3.5.4 Maintainability

This application has minimal number of user interfaces, so that helps to user to use this application very easily and without any conflict. In this application, more services are run on the backend. It helps user to easily maintain the application in user side.

3.6 Other Requirements

3.6.1 Interoperability

This application not contains any interoperability ability to connect with other software or any external service.