GNG1103

Design Project User and Product Manual

Sim Swapping Prevention by Location and Quality of Service Reduction

Submitted by:

SIM Tech, Group 17

Pavithra Raj Mohan, 300363388

Stéphane Lauzon-Brisson, 300009543

Nour Mokdad, 300447978

Kushal Raveen Jayarathna

Tuesday, December 3, 2024

University of Ottawa

Table of Contents

Table of Contents ii

List of Tables iv

List of Acronyms and Glossary v

1 Introduction 1

2 Overview 2

2.1 Conventions 3

2.2 Cautions & Warnings 3

3 Getting started 4

3.1 Configuration Considerations 4

3.2 User Access Considerations 4

3.3 Accessing/setting up the System 4

3.4 System Organization & Navigation 5

3.5 Exiting the System 5

4 Using the System 6

4.1 QoS Reduction to Simulate SIM Swap 6

4.1.1 QoS Reinstating or Maintenance Based on Administrative Control 6

5 Troubleshooting & Support 6

5.1 Error Messages or Behaviors 6

5.2 Special Considerations 6

5.3 Maintenance 6

5.4 Support 7
6 Product Documentation 8
6.1 <Subsystem 1 of prototype> 8
6.1.1 BOM (Bill of Materials) 8
6.1.2 Equipment list 8
6.1.3 Instructions 8
6.2 Testing & Validation 9
7 Conclusions and Recommendations for Future Work 11
APPENDICES 12
8 APPENDIX I: Design Files 12

List of Tables

Table 1. Acronyms v

Table 2. Glossary v

Table 3. Referenced Documents 12

List of Acronyms and Glossary

Table 1. Acronyms

Acronym	Definition
GUI	Graphical User Interface (parts of the application that users interact with like buttons, etc.)
SMS	SMS stands for Short Message Service and is used to send short text messages (less than 160 characters) between cellular networks.

Table 2. Glossary

Term	Acronym	Definition
Application Programming Interface	API	A set of rules allowing software applications to communicate together. They define how they interact using requests and responses.
Internet of Things	IoT	Devices that are connected and can communicate between themselves and the cloud.
Quality of Service	QoS	Technology that manages network traffic to ensure prioritization of applications, reducing issues like network jitter, packet loss, and high latency.
Subscriber Identity Module	SIM	A card that contains an identification number matching its device and prevents operation if removed.

1 Introduction

This User and Product Manual (UPM) provides the information necessary for developers and company administrators to effectively use the SIM Swap Preventor (SSP) application and for prototype documentation.

SIM swapping is a form of identity theft having damaging consequences like account take over, identity theft, privacy invasion, legal issues, etc. The main goal of a SIM swap is to gain access to a victim's two factor authentication (2FA) that is sent through SMS which can then grant access to the victim's bank information, etc.

This product focuses on SIM swap attacks targeting Internet of Things (IoT) devices. Such devices can include smart meters, smart vehicles, industrial sensors, security systems, etc., through which attackers can gain unauthorized access to the device's sensitive data and functionality as well.

The SSP offers methods to control such issues in business settings allowing an administrator to be able to be alerted when a fraudulent SIM swap occurs and is able to accept and deny changes. This document can also be used by developers to download and set up the prototype and then adapt it to their own systems.

2 Overview

IoT devices like medical trackers, smart meters, smart vehicles, etc., use IoT SIM cards to wirelessly communicate over mobile networks (3G, 4G, 5G). IoT SIM swapping can expose many types of a business's sensitive data, depending on the device exposed to fraud. For example, employee information, patented documents and methods, security system configurations and information, etc., can be breached. The attacker can gain control of the device as well, which can be detrimental to a business if it is used to redirect communication, remotely disable or enable systems, or cause financial loss.

To prevent such attacks, it is required for a system to immediately stop communication with a device that has been breached, by reducing the quality of service (QoS) and contacting the responsible administrator to minimize the amount of breached information as soon as possible. The SSP reduces the quality of service such that the only communication from the breached device can be a signal to the SSP to monitor the status of the device.

This method of protection ensures that if a SIM swap occurs, the device is protected in contrast to other swap prevention methods such as encryption SIM card locking, multifactor authentication, etc., which do not protect the device in the case of a breach.

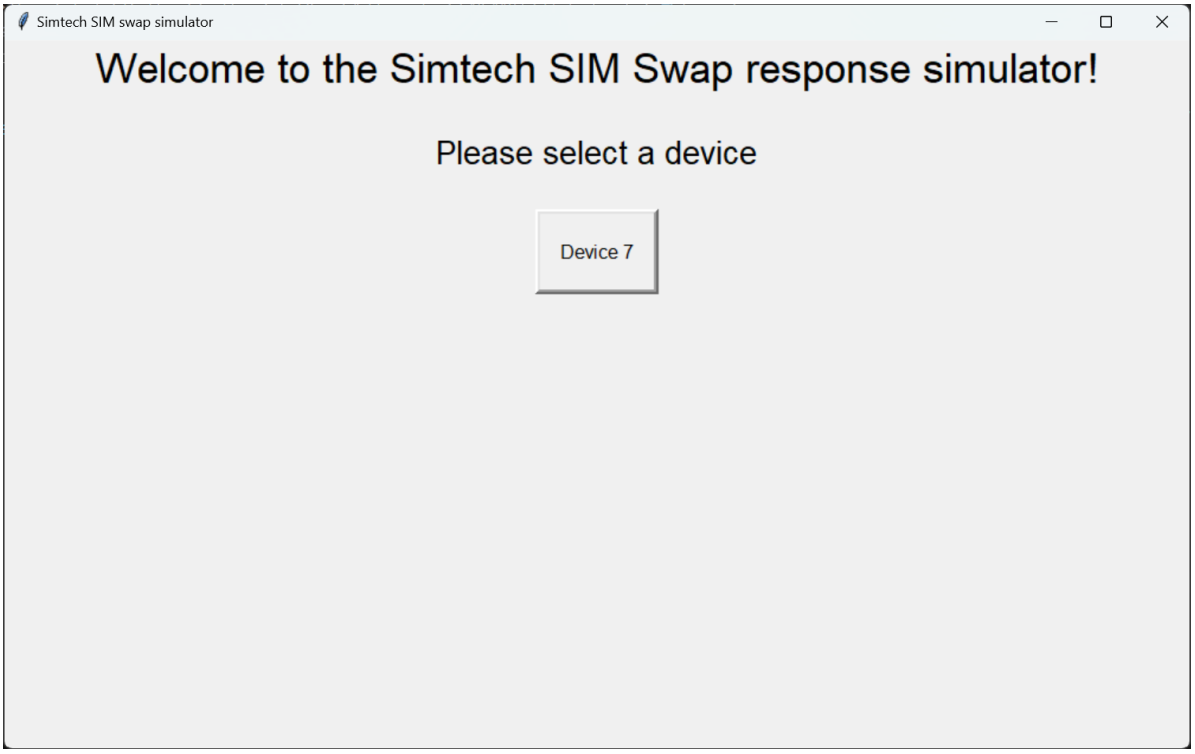

Figure 1: User interface of SIM Swap Protection Simulator

The SSP is simulation run on computers, allowing user access through a graphical user interface (GUI) and text.

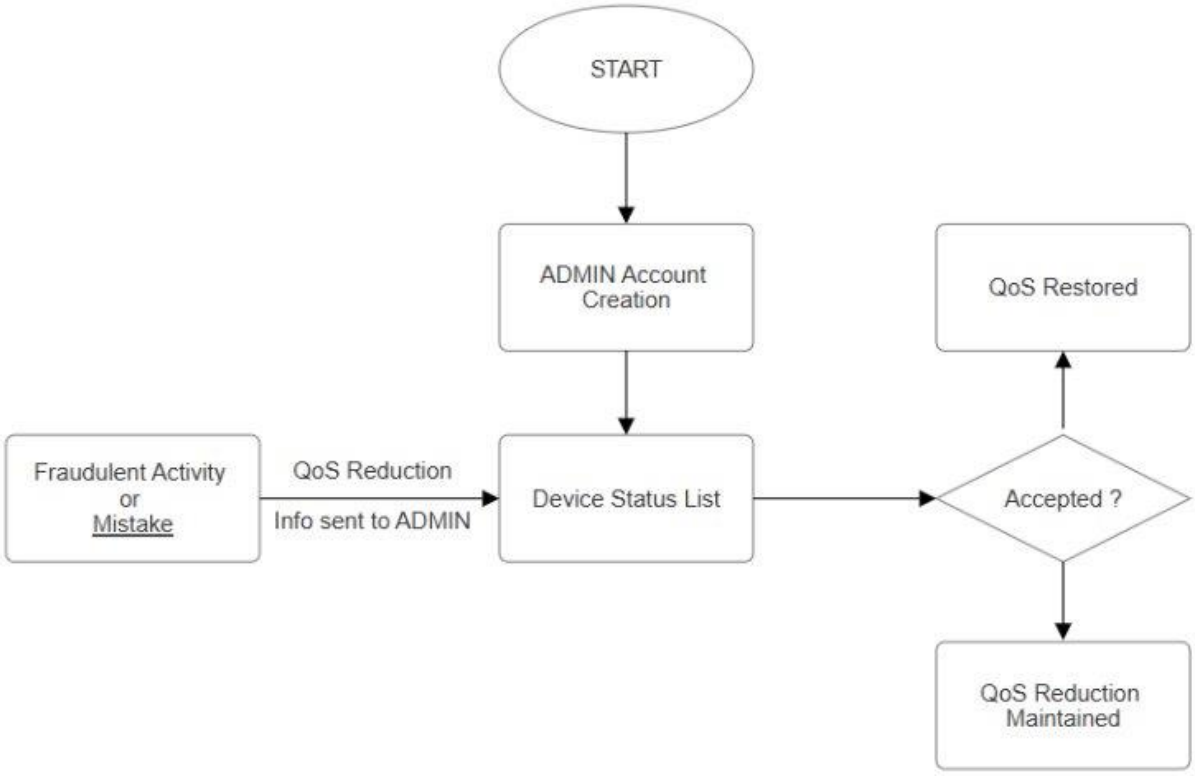

Figure 2: Algorithm Block Diagram

2.1 Conventions

Information pertaining to software download and set up will be titled “Action”. All other information will pertain to information regarding the prototype development and background information.

2.2 Cautions & Warnings

There is currently no method of encryption on this software and using this prototype on a device with access to sensitive information is highly discouraged.

3 Getting started

The SSP runs using a Python compiler and requires the user click on their required device to simulate a SIM swap and begin the quality-of-service reduction process.

3.1 Configuration Considerations

The prototype requires Visual Studio Code or some Python compiler in order to run the program. The Shabodi NetAware API must also be accessed by downloading a VPN and using log-in credentials to access the remote device.

3.2 User Access Considerations

Main users will be the administrators and network engineers so setting up all required devices will be required. As of right now, a SIM swap is simulated with the click of a button as the program is running, so during set-up, network engineers must provide a code/SIM number for each device and SIM being used that the system can continuously compare the values and make sure they are matching up.

If not matching, the system will begin the QoS reduction and alert the administration. Administration staff will be required to be updated and trained on fraudulent activity response and understand how to reinstate the QoS when the system is activated on accident or for other required reasons.

The program itself is very user friendly and is suitable for any working professional regardless of academic background to be able to use and understand the prototype after it has been set up by a networking professional and adequate training has been received.

3.3 Accessing/setting up the System

To access the current prototype the following steps must be followed:

1. Set-up the Shabodi NetAware Sandbox.
2. Download the FortiClient VPN application onto the required computer using https://www.fortinet.com/support/product-downloads
3. Set-up a VPN connection using Shabodi credentials.
4. Download Visual Studio or other types of Python compilers.
5. Download the code for the prototype from the Makerepo project repository
6. Open the GUI code and Design Day Code on a Python compiler (Visual Studio Code)
    a. File > Open > GUI code

    b. Repeat for design day code
7. Navigate to the GUI code on the top tabs bar
8. Click on run (a play button on the top right corner of the window).
9. A terminal should open up allowing the user to view the prototype.

3.4 System Organization & Navigation

Upon running, the program opens to the administrator's home page where all devices can be viewed. Fraudulent activity on the devices will be notified to the administrator and allow for QoS reduction. This stage is simulated by the clicking of the Device 7 button.

Upon SIM swap, the administration is notified of the QoS change and is allowed to either accept or reject the swap (accepting means that the swap was premediated and was not a fraud). Denying the swap gives the administration the power to reinstate the QoS service.

3.5 Exiting the System

Click on the close (X) button at the top right corner of the window to exit the application. Make sure to close the VPN connection as well by navigating to the FortiClient VPN window and clicking "Disconnect".

4 Using the System

The following sub-sections provide detailed, step-by-step instructions on how to use the various functions or features of the SSP application.

4.1 QoS Reduction to Simulate SIM Swap

To simulate a SIM Swap, click on the button named Device 7. When clicked the QoS will be directly reduced, and a message will confirm this feature.

4.1.1 QoS Reinstating or Maintenance Based on Administrative Control

Once reduced, the administrator has two options:
1. Accept the swap: This means that the swap came from within the business and is not a threat. This button will reinstate the QoS.
2. Deny the swap: This button will maintain the reduced QoS and advise the administrator to take serious action like police, higher authorities, etc.

5 Troubleshooting & Support

In case of an error, make sure to check the VPN connection and reconnect with your credentials.

5.1 Error Messages or Behaviors

TimeoutError: [WinError 10060] A connection attempt failed because the connected party did not properly respond after a period, or established connection failed because connected host has failed to respond

If Python code does not run or the following error shows up, refer to the above and recheck the VPN connection and re-enter the log-in credentials.

5.2 Special Considerations

In some networks connected to Wi-Fi, the VPN may not work due to restrictions and using a cellular network (hotspot) may be required.

5.3 Maintenance

No maintenance is required for this prototype.

5.4 Support

All support may come from developer.support@shabodi.com with regards to VPN and access issues or API issues.

6 Product Documentation

6.1 <Subsystem 1 of prototype>

6.1.1 BOM (Bill of Materials)

Item	Purpose	Cost per Unit ($)	Quantity Subtotal ($)
Shabodi NetAware Sandbox	Needed to learn and utilize Shabodi's APIs	0$	0$
Python 3.12.7	Needed to code the subsystems of our product	0$	0$
Microsoft VSCode	Needed to code to code in python outside of the NetAware Sandbox	0$	0$
Tkinter GUI library	GUI library for python that is easy to use and access and has a lot of documentation and tutorials	0$	0$

6.1.2 Equipment list

Computer with access to internet.

6.1.3 Instructions

To gain access to the Shabodi NetAware API, the program must use the token API to generate a key.

The next step is to set a bandwidth for the required device. Set up device bandwidth codes to compare the bandwidth codes to reduce the QoS using an if statement. Changing the QoS should send a message to the user on the GUI.

Creating the GUI is simply creating a window and button that signifies the SIM swap. On click, the button should change the bandwidth code number to initiate the QoS reduction. All of these files

6.2 Testing & Validation

Testing was done with each prototype based on each focus. The first prototype was based on location, second on the snapshot, and third on the QoS reduction. The comprehensive prototype will do a full analysis on the final prototype combining all of the above factors.

Prototype	**Test Objective**	**Description of prototype and test method**	**Results**	**Estimated test duration**
Prototype I	Test snapshot location precision	Request the location of device via Location API multiple times and compare results for precision	Quantitative: Longitude + Latitude coordinates of device	45 minutes
Prototype II	Test snapshot info accuracy	Request device info via API multiple times and compare results accuracy	Quantitative: Snapshot info (time, UE ID) can be compared to actual values	45 minutes
Prototype III	QoS reduction efficiency testing	Trigger a QoS reduction and determine if the	Quantitative: Compare bitrate to the bitrate that we	30 minutes

		bitrate matches the required reduction	are supposed to be receiving after QoS	

7 Conclusions and Recommendations for Future Work

This project could be improved by fine-tuning the location tracking feature. Currently, location is manually entered and using the geolocation API from Google or Shabodi can enhance the overall SIM swap detection. Additionally, the SIM swap detection can be updated to use SIM numbers and device IDs to detect the change in SIM.

The GUI can also be changed to support Shabodi's branding using company colors and formatting.

APPENDICES

8 APPENDIX I: Design Files

Maker-Repo Repository: https://makerepo.com/PavithraRM/2212.gng1103g17simtech

Table 3. Referenced Documents

Document Name	Document Location and/or URL	Issuance Date
Deliverable D	https://makerepo.com/PavithraRM/2212.gng1103g17simtech	10/2024
Deliverable E	https://makerepo.com/PavithraRM/2212.gng1103g17simtech	10/2024
Deliverable F	https://makerepo.com/PavithraRM/2212.gng1103g17simtech	11/2024
Deliverable G	https://makerepo.com/PavithraRM/2212.gng1103g17simtech	11/2024
Deliverable H	https://makerepo.com/PavithraRM/2212.gng1103g17simtech	11/2024