Vencore SecureIO Security Target

Acumen Security, LLC.

Document Version: 0.5

Table Of Contents

Revision History

Version	Date	Description
0.1	7/20/17	Initial Draft
0.2	8/31/17	Updated to address validator comments
0.3	9/18/17	Updated to list TD0119 as relevant
0.4	2/16/18	Updated to address validator comments
0.5	3/6/2018	Updated to address validator comments

1 Security Target Introduction

1.1 Security Target and TOE Reference

This section provides information needed to identify and control this ST and its TOE.

Category	Identifier
ST Title	Vencore SecureIO Security Target
ST Version	0.5
ST Date	March 2018
ST Author	Acumen Security, LLC.
TOE Identifier	Vencore Secure IO
TOE Software Version	1.0
TOE Developer	Vencore Labs
Key Words	TLS Proxy

Table 1 TOE/ST Identification

1.2 TOE Overview

The SecureIO application provides a secure communication channel for Android applications to send and receive network traffic. The traffic will be protected in transit using TLS from the Android device to a TLS server.

The functionality of the SecureIO service is limited to (i) establishing and shutting down a TLS connection to the Transport Layer Gateway (TLG); (ii) sending and receiving messages to and from the TLG on behalf of Android apps via the TLS connection.

1.3 TOE Architecture

1.3.1 Physical Boundaries

The TOE is a software application that resides entirely on its Android-based mobile platform.

1.3.2 Security Functions provided by the TOE

The TOE provides the security functionality required by [SWAPP].

1.3.2.1 Cryptographic Support

The TOE relies on underlying cryptographic functionality provided by the platform for all of its cryptographic operations.

1.3.2.2 User Data Protection

The TOE is a TLS proxy that encrypts data sent by other applications on its host platform.

1.3.2.3 Security Management

The TOE does not come with any default credentials. It identifies itself to the TLS gateway that it connects to using a certificate and private key. These are provisioned onto the TOE by an administrator or end user.

1.3.2.4 Privacy

The TOE itself does not contain or transmit any PII. It functions as a TLS proxy over which other applications on the platform may transmit whatever data they wish.

1.3.2.5 Protection of the TSF

The TOE employs several mechanisms to ensure that it is secure on the host platform. Only documented platform APIs are used by the TOE. The TOE never allocates memory with both write and execute permission. Evaluated platform functionality is used to verify the TOE version and perform updates, and no third-party libraries are used.

1.3.2.6 Trusted Path/Channels

TLS is used to protect all data transmitted to and from the TOE.

1.3.3 TOE Documentation

- [ST] Vencore SecureIO Security Target, Version 0.5
- [AGD] SecureIO User Manual, Version 1.2

1.3.4 Other References

Protection Profile for Application Software, version 1.2, dated, 22 April 2016 [SWAPP].

1.3.5 TOE Environmental Requirements

The TOE runs on Android versions 6.0, 7.0, and 7.1. All sub-versions (e.g. 6.0.1) of 6.0 and 7.0 are supported. The TOE was tested using the CC validated versions of the Android OS including,

- Samsung Galaxy Devices on Android 7.1 (VID10849)
- Samsung Galaxy Devices with Android 7 (VID10809)
- Samsung Galaxy Devices with Android 6 (VID10726)

2 Conformance Claims

2.1 CC Conformance

This TOE is conformant to:

- Common Criteria for Information Technology Security Evaluations Part 1, Version 3.1, Revision 4, September 2012
- Common Criteria for Information Technology Security Evaluations Part 2, Version 3.1, Revision 4, September 2012: Part 2 extended
- Common Criteria for Information Technology Security Evaluations Part 2, Version 3.1, Revision 4, September 2012: Part 3 extended

2.2 Protection Profile Conformance

This TOE is conformant to:

- Protection Profile for Application Software, version 1.2, dated, 22 April 2016 [SWAPP].

2.3 Conformance Rationale

This Security Target provides exact conformance to Version 1.2 of the Protection Profile for Application Software. The security problem definition, security objectives and security requirements in this Security Target are all taken from the Protection Profile performing only operations defined there.

2.3.1 Technical Decisions

The following Technical Decisions have been considered for this evaluation:

- TD0119: FCS_STO_EXT.1.1 in PP_APP_v1.2: This TD applies to products which implement this functionality. The TOE invokes platform functionality for this requirement.
- TD0163: Updated to FCS_TLSC_EXT.1.1 Test 5.4 and FCS_TLSS_EXT.1.1 Test 4.3
- TD0174: Optional Ciphersuites for TLS
- TD0192: Updated to FCS_STO_EXT.1 Application Note
- TD0217: Compliance to RFC5759 and RFC5280 for using CRLs
- TD0221: FMT_SMF.1.1 – Assignments moved to selections
- TD0238: User-modifiable files FPT_AEX_EXT.1.4
- TD0244: FCS_TLSC_EXT – TLS Client Curves Allowed
- TD0283: Cipher Suites for TLS in SWAPP v1.2

2.3.2 Non-Applicable Technical Decisions

The following Technical Decisions have bene issued for this PP but were found not to apply to this evaluation.

- TD0107: FCS_CKM - ANSI X9.31-1998, Section 4.1.for Cryptographic Key Generation: FCS_CKM.1 is a selection based SFR that is not included in this ST.

- TD0121: FMT_MEC_EXT.1.1 Configuration Options: This TD applies to products on a Windows platform which claim compliance to the File Encryption EP. The TOE meets neither of those criteria.
- TD0122: FMT_SMF.1.1 Assignments Moved to Selections: TD has been archived and replaced by TD0221.
- TD0131: Updated to FCS_TLSS_EXT.1.1 Test 4.5: The TOE does not implement TLSS functionality.
- TD0172: Additional APIs added to FCS_RBG_EXT.1.1: This TD only applies to TOEs on a Windows platform.
- TD0177: FCS_TLSS_EXT.1 Application Note Update: The TOE does not implement TLSS functionality.
- TD0178: Integrity for installation tests in AppSW PP: This TD only applies to TOEs on iOS platforms.
- TD0215: Update to FCS_HTTPS_EXT.1.2: The TOE does not implement HTTPS functionality. 0218
- TD0218: Update to FPT_AEX_EXT.1.3 Assurance Activity: This TD only applies to TOEs on a Windows platform.
- TD0241: TOE is not a TLS server.
- TD0267: TOE is not a TLS server.
- TD0268: TOE does not remotely store configuration data.
- TD0269: TOE does not run on Windows
- TD0293: FCS_CKM.1(1) is not selected in the ST.

3 Security Problem Definition

The security problem definition has been taken from [SWAPP] and is reproduced here for the convenience of the reader. The security problem is described in terms of the threats that the TOE is expected to address, assumptions about the operational environment, and any organizational security policies that the TOE is expected to enforce.

3.1 Threats

The following threats are drawn directly from the SWAPP.

ID	Threat
T.NETWORK_ATTACK	An attacker is positioned on a communications channel or elsewhere on the network infrastructure. Attackers may engage in communications with the application software or alter communications between the application software and other endpoints in order to compromise it.
T.NETWORK_EAVESDROP	An attacker is positioned on a communications channel or elsewhere on the network infrastructure. Attackers may monitor and gain access to data exchanged between the application and other endpoints.
T.LOCAL_ATTACK	An attacker can act through unprivileged software on the same computing platform on which the application executes. Attackers may provide maliciously formatted input to the application in the form of files or other local communications.
T.PHYSICAL_ACCESS	An attacker may try to access sensitive data at rest.

Table 2 Threats

3.2 Assumptions

The following assumptions are drawn directly from the SWAPP.

ID	Assumption
A.PLATFORM	The TOE relies upon a trustworthy computing platform for its execution. This includes the underlying platform and whatever runtime environment it provides to the TOE.
A.PROPER_USER	The user of the application software is not willfully negligent or hostile, and uses the software in compliance with the applied enterprise security policy.
A.PROPER_ADMIN	The administrator of the application software is not careless, willfully negligent or hostile, and administers the software within compliance of the applied enterprise security policy.

Table 3 OSPs

3.3 Organizational Security Policies

There are no OSPs for the application

4 Security Objectives

The security objectives have been taken from [SWAPP] and are reproduced here for the convenience of the reader.

4.1 Security Objectives for the TOE

The following security objectives for the TOE were drawn directly from the SWAPP.

ID	TOE Objective
O.INTEGRITY	Conformant TOEs ensure the integrity of their installation and update packages, and also leverage execution environment-based mitigations. Software is seldom if ever shipped without errors, and the ability to deploy patches and updates to fielded software with integrity is critical to enterprise network security. Processor manufacturers, compiler developers, execution environment vendors, and operating system vendors have developed execution environment-based mitigations that increase the cost to attackers by adding complexity to the task of compromising systems. Application software can often take advantage of these mechanisms by using APIs provided by the runtime environment or by enabling the mechanism through compiler or linker options. Addressed by: FDP_DEC_EXT.1, FMT_CFG_EXT.1, FPT_AEX_EXT.1, FPT_TUD_EXT.1
O.QUALITY	To ensure quality of implementation, conformant TOEs leverage services and APIs provided by the runtime environment rather than implementing their own versions of these services and APIs. This is especially important for cryptographic services and other complex operations such as file and media parsing. Leveraging this platform behavior relies upon using only documented and supported APIs. Addressed by: FMT_MEC_EXT.1, FPT_API_EXT.1, FPT_LIB_EXT.1
O.MANAGEMENT	To facilitate management by users and the enterprise, conformant TOEs provide consistent and supported interfaces for their security-relevant configuration and maintenance. This includes the deployment of applications and application updates through the use of platform-supported deployment mechanisms and formats, as well as providing mechanisms for configuration. This also includes providing control to the user regarding disclosure of any PII. Addressed by: FMT_SMF.1, FPT_IDV_EXT.1, FPT_TUD_EXT.1.5, FPR_ANO_EXT.1
O.PROTECTED_STORAGE	To address the issue of loss of confidentiality of user data in the event of loss of physical control of the storage medium, conformant TOEs will use data-at-rest protection. This involves encrypting data and keys stored by the TOE in order to prevent unauthorized access to this data. This also includes unnecessary network communications whose consequence may be the loss of data. Addressed by: FDP_DAR_EXT.1, FCS_STO_EXT.1, FCS_RBG_EXT.1
O.PROTECTED_COMMS	To address both passive (eavesdropping) and active (packet modification) network attack threats, conformant TOEs will use a trusted channel for sensitive data. Sensitive data includes cryptographic keys, passwords, and any other data specific to the application that should not be exposed outside of the application. Addressed by: FTP_DIT_EXT.1, FCS_TLSC_EXT.1, FCS_DTLS_EXT.1, FCS_RBG_EXT.1

Table 4 Objectives for the TOE

4.2 Security Objectives for the Operational Environment

The following security objectives for the operational environment assist the TOE in correctly providing its security functionality. These track with the assumptions about the environment.

ID	Objective for the Operation Environment
OE.PLATFORM	The TOE relies upon a trustworthy computing platform for its execution. This includes the underlying operating system and any discrete execution environment provided to the TOE.
OE.PROPER_USER	The user of the application software is not willfully negligent or hostile, and uses the software within compliance of the applied enterprise security policy.
OE.PROPER_ADMIN	The administrator of the application software is not careless, willfully negligent or hostile, and administers the software within compliance of the applied enterprise security policy.

Table 5 Objectives for the environment

5 Security Requirements

This section identifies the Security Functional Requirements for the TOE and/or Platform. The Security Functional Requirements included in this section are derived from Part 2 of the Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 4, dated: September 2012 and all international interpretations.

Requirement	Requirement Description
Mandatory SFRs	
FCS_RBG_EXT.1	Random Bit Generation Services
FCS_STO_EXT.1	Storage of Secrets
FDP_DEC_EXT.1	Access to Platform Resources
FDP_NET_EXT.1	Network Communications
FDP_DAR_EXT.1	Encryption Of Sensitive Application Data
FMT_MEC_EXT.1	Supported Configuration Mechanism
FMT_CFG_EXT.1	Secure by Default Configuration
FMT_SMF.1	Specification of Management Functions
FPR_ANO_EXT.1	User Consent for Transmission of Personally Identifiable Info
FPT_API_EXT.1	Use of Supported Services and APIs
FPT_AEX_EXT.1	Anti-Exploitation Capabilities
FPT_TUD_EXT.1	Integrity for Installation and Update
FPT_LIB_EXT.1	Use of Third Party Libraries
FTP_DIT_EXT.1	Protection of Data in Transit
Optional, Selection-Based and Objective SFRs	
FCS_TLSC_EXT.1	TLS Client Protocol
FCS_TLSC_EXT.2	TLS Client Protocol
FCS_TLSC_EXT.4	TLS Client Protocol
FIA_X509_EXT.1	X.509 Certificate Validation
FIA_X509_EXT.2	X.509 Certificate Authentication

Table 6 SFRs

5.1 Conventions

The CC defines operations on Security Functional Requirements: assignments, selections, assignments within selections and refinements. This document uses the following font conventions to identify the operations defined by the CC:

- Assignment: Indicated with *italicized* text;
- Refinement: Indicated with **bold** text;
- Selection: Indicated with underlined text;
- Iteration: Indicated by appending the iteration number in parenthesis, e.g., (1), (2), (3).
- Where operations were completed in the PP itself, the formatting used in the PP has been retained.

Explicitly stated SFRs are identified by having a label ‘EXT’ after the requirement name for TOE SFRs. Formatting conventions outside of operations matches the formatting specified within the PP.

5.2 Security Functional requirements

5.2.1 Cryptographic Support (FCS)

FCS_RBG_EXT.1 Random Bit Generation Services

FCS_RBG_EXT.1.1

The application shall use no DRBG functionality for its cryptographic operations

FCS_STO_EXT.1 Storage of Secrets

FCS_STO_EXT.1.1

The application shall invoke the functionality provided by the platform to securely store *X.509 certificates* to non-volatile memory.

FCS_TLSC_EXT.1 TLS Client Protocol

FCS_TLSC_EXT.1.1
The application shall invoke platform-provided TLS 1.2 supporting the following cipher suites:

- TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 5246
- TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246,
- TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246,
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA_256 as defined in RFC 5246,
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA_256 as defined in RFC 5246,
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289
- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289
- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289,
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289,
- TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289,
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289

and no other cipher suite.

FCS_TLSC_EXT.1.2

The application shall verify that the presented identifier matches the reference identifier according to RFC 6125.

FCS_TLSC_EXT.1.3

The application shall establish a trusted channel only if the peer certificate is valid.

FCS_TLSC_EXT.2 TLS Client Protocol

FCS_TLSC_EXT.2.1

The application shall support mutual authentication using X.509v3 certificates.

FCS_TLSC_EXT.4 TLS Client Protocol

FCS_TLSC_EXT.4.1

The application shall present the supported Elliptic Curves Extension in the Client Hello with the following NIST curves: secp256r1, secp384r1, secp521r1.

Application Note: When installed on Samsung Galaxy Devices with Android 7 (VID10809) the connection will only support secp256r1 elliptic curve. This is because of a limitation in the Android OS. Google has acknowledged that this is a bug, but has tagged it "*will not fix*."

5.2.2 User Data Protection (FDP)

FDP_DEC_EXT.1 Access to Platform Resources

FDP_DEC_EXT.1.1

The application shall restrict its access to network connectivity.

FDP_DEC_EXT.1.2

The application shall restrict its access to no sensitive information repositories.

FDP_NET_EXT.1 Network Communications

FDP_NET_EXT.1.1

The application shall restrict network communication to user-initiated communication for *secure tunnel establishment*.

FDP_DAR_EXT.1 Encryption Of Sensitive Application Data

FDP_DAR_EXT.1.1

The application shall not store any sensitive data in non-volatile memory.

5.2.3 Identification and Authentication (FIA)

FIA_X509_EXT.1 X.509 Certificate Validation

FIA_X509_EXT.1.1

The application shall invoke platform-provided functionality to validate certificates in accordance with the following rules:

- RFC 5280 certificate validation and certificate path validation.
- The certificate path must terminate with a trusted CA certificate.
- The application shall validate a certificate path by ensuring the presence of the basicConstraints extension and that the CA flag is set to TRUE for all CA certificates.

- The application shall validate the revocation status of the certificate using [the Online Certificate Status Protocol (OCSP) as specified in RFC 2560, a Certificate Revocation List (CRL) as specified in RFC 5759].
- The application shall validate the extendedKeyUsage field according to the following rules:
    - Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.
    - Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.
    - Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field.
    - S/MIME certificates presented for email encryption and signature shall have the Email Protection purpose (id-kp 4 with OID 1.3.6.1.5.5.7.3.4) in the extendedKeyUsage field.
    - OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.
    - Server certificates presented for EST shall have the CMC Registration Authority (RA) purpose (id-kp-cmcRA with OID 1.3.6.1.5.5.7.3.28) in the extendedKeyUsage field.

FIA_X509_EXT.1.2

The application shall only treat a certificate as a CA certificate if the basicConstraints extension is present and the CA flag is set to TRUE.

FIA_X509_EXT.2 X.509 Certificate Authentication

FIA_X509_EXT.2.1

The application shall use X.509v3 certificates as defined by RFC 5280 to support authentication for TLS.

FIA_X509_EXT.2.2

When the application cannot establish a connection to determine the validity of a certificate, the application shall not accept the certificate.

5.2.4 Security Management (FMT)

FMT_MEC_EXT.1 Supported Configuration Mechanism

FMT_MEC_EXT.1.1

The application shall invoke the mechanisms recommended by the platform vendor for storing and setting configuration options.

FMT_CFG_EXT.1 Secure by Default Configuration

FMT_CFG_EXT.1.1

The application shall only provide enough functionality to set new credentials when configured with

default credentials or no credentials.

FMT_CFG_EXT.1.2

The application shall be configured by default with file permissions which protect it and its data from unauthorized access.

FMT_SMF.1 Specification of Management Functions

FMT_SMF.1.1

The TSF shall be capable of performing the following management functions

- no other function.

5.2.5 Privacy (FPR)

FPR_ANO_EXT.1 User Consent for Transmission of Personally Identifiable Information

FPR_ANO_EXT.1.1

The application shall not transmit PII over a network.

5.2.6 Protection of TSF (FPT)

FPT_API_EXT.1 Use of Supported Services and APIs

FPT_API_EXT.1.1

The application shall only use documented platform APIs.

FPT_AEX_EXT.1 Anti-Exploitation Capabilities

FPT_AEX_EXT.1.1

The application shall not request to map memory at an explicit address except for *no exceptions*.

FPT_AEX_EXT.1.2

The application shall not allocate any memory region with both write and execute permissions.

FPT_AEX_EXT.1.3

The application shall be compatible with security features provided by the platform vendor.

FPT_AEX_EXT.1.4

The application shall not write user-modifiable files to directories that contain executable files unless explicitly directed by the user to do so.

FPT_AEX_EXT.1.5

The application shall be compiled with stack-based buffer overflow protection enabled.

FPT_TUD_EXT.1 Integrity for Installation and Update

FPT_TUD_EXT.1.1

The application shall leverage the platform to check for updates and patches to the application

software.

FPT_TUD_EXT.1.2

The application shall be distributed using the format of the platform-supported package manager.

FPT_TUD_EXT.1.3

The application shall be packaged such that its removal results in the deletion of all traces of the application, with the exception of configuration settings, output files, and audit/log events.

FPT_TUD_EXT.1.4

The application shall not download, modify, replace or update its own binary code.

FPT_TUD_EXT.1.5

The application shall provide the ability to query the current version of the application software.

FPT_TUD_EXT.1.6

The application installation package and its updates shall be digitally signed such that its platform can cryptographically verify them prior to installation.

FPT_LIB_EXT.1 Use of Third Party Libraries

FPT_LIB_EXT.1.1

The application shall be packaged with only *no third-party libraries*.

5.2.7 Trusted Path/Channel (FTP)

FTP_DIT_EXT.1 Protection of Data in Transit

FTP_DIT_EXT.1.1

The application shall encrypt all transmitted sensitive data with TLS between itself and another trusted IT product.

5.3 TOE SFR Dependencies Rationale for SFRs

The Protection Profile for Application Software contains all the requirements claimed in this Security Target. As such, the dependencies are not applicable since the PP has been approved.

5.4 Security Assurance Requirements

The TOE assurance requirements for this ST are taken directly from the Protection Profile for Application Software which are derived from Common Criteria Version 3.1, Revision 4. The assurance requirements are summarized in the table below.

Assurance Class	Components	Components Description
Development	ADV_FSP.1	Basic Functional Specification
Guidance Documents	AGD_OPE.1	Operational User Guidance
	AGD_PRE.1	Preparative User Guidance
Life Cycle Support	ALC_CMC.1	Labeling of the TOE
	ALC_CMS.1	TOE CM Coverage

Assurance Class	Components	Components Description
	ALC_TSU_EXT.1	Timely Security Updates
Tests	ATE_IND.1	Independent Testing – Conformance
Vulnerability Assessment	AVA_VAN.1	Vulnerability Analysis

Table 7 Security Assurance Requirements

5.5 Rationale for Security Assurance Requirements

The functional specification describes the external interfaces of the TOE; such as the means for a user to invoke a service and the corresponding response of those services. The description includes the interface(s) that enforces a security functional requirement, the interface(s) that supports the enforcement of a security functional requirement, and the interface(s) that does not enforce any security functional requirements. The interfaces are described in terms of their purpose (general goal of the interface), method of use (how the interface is to be used), parameters (explicit inputs to and outputs from an interface that control the behavior of that interface), parameter descriptions (tells what the parameter is in some meaningful way), and error messages (identifies the condition that generated it, what the message is, and the meaning of any error codes). The development evidence also contains a tracing of the interfaces to the SFRs described in this ST.

5.6 Assurance Measures

The TOE satisfies the identified assurance requirements. This section identifies the Assurance Measures applied by the vendor to satisfy the assurance requirements. The table below lists the details.

SAR Component	How the SAR will be met
ADV_FSP.1	The functional specification describes the external interfaces of the TOE; such as the means for a user to invoke a service and the corresponding response of those services. The description includes the interface(s) that enforces a security functional requirement, the interface(s) that supports the enforcement of a security functional requirement, and the interface(s) that does not enforce any security functional requirements. The interfaces are described in terms of their purpose (general goal of the interface), method of use (how the interface is to be used), parameters (explicit inputs to and outputs from an interface that control the behavior of that interface), parameter descriptions (tells what the parameter is in some meaningful way), and error messages (identifies the condition that generated it, what the message is, and the meaning of any error codes).
AGD_OPE.1	The Administrative Guide provides the descriptions of the processes and procedures of how the administrative users of the TOE can securely administer the TOE using the interfaces that provide the features and functions detailed in the guidance.
AGD_PRE.1	The Installation Guide describes the installation, generation, and startup procedures so that the users of the TOE can put the components of the TOE in the evaluated configuration.
ALC_CMC.1	The Configuration Management (CM) documents describe how the consumer identifies the evaluated TOE. The CM documents identify the configuration items, how those configuration items are uniquely identified, and the adequacy of the procedures that are used to control and track changes that are made to the TOE. This includes details on what changes are tracked and how potential changes are incorporated.
ALC_CMS.1	
ALC_TSU_EXT.1	Vencore uses a systematic method for identifying and providing security relevant updates to the TOEs users via its support infrastructure.
ATE_IND.1	Vencore will provide the TOE for testing.
AVA_VAN.1	Vencore will provide the TOE for testing.

Table 8 TOE Security Assurance Measures

6 TOE Summary Specification

This chapter identifies and describes how the Security Functional Requirements identified above are met by the TOE.

Requirement	Rationale
FCS_RBG_EXT.1	The TOE does not use DRBG functionality for its cryptographic operations Due to leveraging of platform cryptographic functionality there are no TOE functions covered by ST SFRs that use random numbers provided by the platform. All random numbers used by SFR related functions are used by the platform's underlying cryptographic functionality.
FCS_STO_EXT.1	Digital certificates (and the keys associated with the digital certificates), which are the secure credentials used for connection authorization by the TOE, are stored within the Android key store on the platform. When needed, the user selects the credentials to use from the platform itself.
FCS_TLSC_EXT.1	The TOE implements TLS 1.2 for use in establishing secure connections to external IT entities. By default, TLS 1.0, TLS 1.1, SSL 2.0 and SSL 3.0 connections are denied. The TOE supports the following encryption algorithms for use with TLS connections: • TLS_RSA_WITH_AES_128_CBC_SHA256 • TLS_RSA_WITH_AES_256_CBC_SHA256 • TLS_DHE_RSA_WITH_AES_128_CBC_SHA_256 • TLS_DHE_RSA_WITH_AES_256_CBC_SHA_256 • TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 • TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 • TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 • TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 • TLS_RSA_WITH_AES_128_CBC_SHA • TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 • TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 • TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 • TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 During establishment of the TLS 1.2 session, the TOE will perform verification of the presented identifier in the peer certificate to ensure that it is a valid reference identifier. This ensures that the reference identifier is conformant with RFC 6125. The TOE supports the following reference identifiers: • IP address • Hostname (DNS) [User Guide p. 14] The TOE does support wildcards and IP addresses. Additionally, the TOE does not perform certificate pinning.
FCS_TLSC_EXT.2	The TOE allows the user to select a certificate to use for identification.

FCS_TLSC_EXT.4	The TOE supports the secp256r1, secp384r1 and secp521r1 NIST curves when it is installed on an Android 6.0 or 7.1.1 platform. When installed on an Android 7.0 platform the TOE is limited to secp256r1. Support for these curves is configured by default when the TOE is installed on a CC evaluated platform.
FDP_DAR_EXT.1	During operation of the TOE, no sensitive data is stored in non-volatile memory. It is not possible for the TOE to store such data because it never receives it.
FDP_DEC_EXT.1	During operation of the TOE, access to the underlying platform is limited to use of network connectivity hardware for establishment of secure communication channels. No sensitive information repositories are accessible.
FDP_NET_EXT.1	During regular operation of the TOE, secure TLS sessions may be established to provide secure channels for communications. These interactions are performed based on the following events: • Pressing the “Connect” button [User guide, p.16]
FIA_X509_EXT.1 FIA_X509_EXT.2	Certificate validation and certificate path validation performed by the TOE platform (Android) is conformant with RFC 5280. The TOE is configured with a single certificate which it uses to identify itself when functioning as a TLS client. Validity checks are performed using functionality implemented in the TOE platform (android). For certificates to successfully validate, the certificate cannot be revoked. Certificate revocation is determined using either a CRL check or OCSP. In addition to the revocation check, the certificate must have a valid basicConstraints extension and extendedKeyUsage field. If for any reason the TOE is unable to determine the validity of a certificate, the certificate will not be accepted.
FMT_CFG_EXT.1	The TOE does not come with any default credentials. It identifies itself to the TLS gateway that it connects to using a certificate and private key. These are provisioned onto the TOE by an administrator or end user.
FMT_MEC_EXT.1 FMT_SMF.1	The TOE maintained a restricted configuration with no management functions being performed by users.
FPR_ANO_EXT.1	The TOE does not transmit any PII over the network.
FPT_AEX_EXT.1	As of Android 5.0 all non-Position Independent Executable (PIE) support was dropped from the Android OS. All processes running on the evaluated Android OS versions (6, 7, and 7.1) have full ASLR support. No additional explicit compiler flags are required to enable ASLR. Additionally, the application is a coded in Java. Java natively checks array bounds.
FPT_API_EXT.1	The TOE leverages the following platform APIs: • java.io.IOException; • java.io.InputStream; • java.net.InetAddress; • java.net.InetSocketAddress; • java.net.SocketException; • java.security.KeyStore; • java.security.PrivateKey; • java.security.PublicKey;

	• java.security.Signature; • java.security.cert.Certificate; • java.security.cert.X509Certificate; • javax.net.ssl.HostnameVerifier; • javax.net.ssl.KeyManager; • javax.net.ssl.KeyManagerFactory; • javax.net.ssl.SSLHandshakeException; • javax.net.ssl.SSLPeerUnverifiedException; • javax.net.ssl.SSLSession; • javax.net.ssl.SSLSocket; • javax.net.ssl.SSLSocketFactory; • javax.net.ssl.TrustManager; • javax.net.ssl.TrustManagerFactory; • javax.net.ssl.X509TrustManager;
FPT_LIB_EXT.1	The TOE does not come packaged with any third-party libraries.
FPT_TUD_EXT.1	The TOE leverages the underlying platform to check for updates and patches to the application software. All updates are packaged in the Android Application Package (APK) format. All updates are digitally signed to ensure they are provided by Vencore, which is the only authorized source for software updates. In the event that any security vulnerability applies to SecureIO, the Vencore Labs will deliver an update within 30 days. Users of the SecureIO app should report any security related issues to the Vencore Labs support team at trusted.csfc@vencorelabs.com.
FTP_DIT_EXT.1	All communication sent between the TOE and any external IT entity is encrypted to protect all transmitted data. This communication is performed over TLS.

Table 9 TOE Summary Specification SFR Description