Security Target

Cellcrypt Classified 2

Ref:	ST001
Ver:	1.1
Date:	2019
Author:	Acumen Security, LLC

Document Control

Version History

Author(s)	Version	Date	Update
Acumen	1.1	19 April 2019	Updated third-party libraries.
Acumen	1.0	12 April 2019	Updated based on ECR comments.
Acumen	0.14	25 March 2019	Updated based on vendor review, TDs, and test results.
Acumen	0.13	30 Jan 2019	Changed document template. Updated based on TDs and test findings.
Acumen	0.12	3 Oct 2018	Updated based on validator feedback
Acumen	0.11	9 Jul 2018	Additional updates from internal review
Acumen	0.10	28 Jun 2018	Updated based on internal review.
Acumen	0.9	26 Jun 2018	Updated based on vendor feedback
Acumen	0.8	18 Jun 2018	Updated to Cellcrypt template and according to Gap analysis findings.
Acumen	0.7	17 Mar 2017	Updated based on validator check-in comments
Acumen	0.6	15 Mar 2017	Updated based on initial validator feedback
Acumen	0.5	6 Feb 2017	Updated based on vendor feedback
Acumen	0.4	20 Jan 2017	Updated for restarted evaluation
Acumen	0.3	12 May 2016	Updated based on initial validator comments
Acumen	0.2	20 Apr 2016	Updated based on vendor feedback
Acumen	0.1	15 Apr 2016	Initial Draft

CONTENTS

List of Tables

1 Security Target Introduction

This section identifies the Security Target (ST) and Target of Evaluation (TOE) identification, ST conventions, ST conformance claims, and the ST organization.

The TOE is comprised of Cellcrypt Classified 2 version 2.10.0, a Secure Voice over Internet Protocol (SVoIP) application for smartphones, which enables users to have secure voice calls on an end-to-end encrypted session.

The Security Target contains the following additional sections:

- Conformance Claims (Section 2)—provides details of conformance of the security target against Common Criteria and provides rationale that the TOE conforms to the PP(s) for which conformance has been claimed.
- Security Problem Definition (Section 3)—specifies the assumptions and threats that define the security problem to be addressed by the TOE and its operational environment.
- Security Objectives (Section 4)—specifies the security objectives for the TOE and its operational environment necessary to counter the threats and satisfy the assumptions defining the security problem.
- Extended Components Definition (Section 5)—specifies any new components that define extended functional and extended assurance requirements.
- TOE Summary Specification (Section 6)—describes the security functions of the TOE and how they satisfy the security functional requirements.

1.1 Security Target Reference

ST Title – Cellcrypt Classified 2 Security Target

ST Version – Version 1.1

ST Date – 19 April 2019

ST Author – Acumen Security, LLC

1.2 TOE Reference

TOE Identification – Cellcrypt Classified 2

TOE Developer – Cellcrypt Inc.

Evaluation Sponsor – Cellcrypt Inc.

CC Identification – Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 4, September 2012

Software Version – 2.10.0

1.3 TOE Overview

Cellcrypt Classified 2 version 2.10.0 is a VOIP application for secure encrypted voice calls designed to run on standard mobile phones. It comprises a handset software application and a back-end support infrastructure (SIP server). Only the handset software application “Cellcrypt Classified 2” is regarded as the TOE.

Cellcrypt Classified uses standard wireless packet-based connectivity that can be provided by a cellular network or a Wi-Fi data connection.

Authenticated connection set-up ensures that only Cellcrypt Classified enabled mobile phones can participate in secure sessions. End-to-end encryption is achieved through the creation and use of session unique encryption keys. These are used by the handset software application when encrypting/decrypting secure voice traffic.

The following assumptions have been made:

- The mobile platform will be a Samsung Galaxy S7 running Android 7.0 on a Snapdragon 820 ARMv8 processor.
- Cellcrypt Classified will only handle single session, unicast, secure voice calls.

Non-TOE requirements:

- Mobile OS (including VPN access), as defined by the Protection Profile for Mobile Device Fundamentals is outside the scope of this evaluation.
- SIP Server, as defined by the SIP Server Extended Package is outside the scope of this evaluation.
- Update check server, webserver hosting the version identifier for the current version of the TOE.
- Configuration server, webserver hosting the inactive call timeout settings.
- Cellcrypt Classified will operate exclusively within the mobility ecosystem specified by the associated mobility Protection Profiles and will assume that all associated resources (IPSEC VPN tunnel, SIP network) are in place.

1.4 TOE Description

The Target of Evaluation (TOE) is the Cellcrypt Classified 2 smartphone application, which will run on an Android 7 based platform. The Cellcrypt Classified 2 application is a software cryptographic application for smartphones, which enables users to have secure voice calls on an end-to-end encrypted session.

The logical scope of the TOE comprises:

- Authenticated connection set-up with a SIP server
- End-to-end encryption used by the TOE when encrypting/decrypting secure voice traffic

The TOE utilizes X.509 Certificates to provide a mutual authentication for the trusted channel with the SIP server. The validity of the X.509 certificates is checked by querying a CRL. The TOE uses the TLSv1.2 protocol to protect all communications with the SIP server from modification and disclosure. In addition to the X.509 Certificate authentication, the TOE authenticates to the SIP server using a password as an additional layer of security. The TOE does not store the password and requires the user to enter the password whenever the TOE requires it.

The TOE achieves end-to-end encryption using SDES-SRTP trusted channel. The keys for the SDES-SRTP trusted channel are protected by the TLS/SIP channel while the keys are being established.

The TOE mitigates side channel attacks by utilizing a fixed rate vocoder. This prevents an attacker from inferring information about the audio based on the bitrate being transmitted. The TOE also enables ASLR and stack-based overflow protections.

2 Conformance Claims

2.1 CC Conformance Claims

The ST and the TOE it describes are conformant to the following CC specifications:

- 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

The TOE claims exact conformance to:

- Protection Profile for Application Software Version 1.2, 2016-04-25 [AppPP]
- Extended Package for Voice and Video over IP (VVoIP) Version 1.0, 2016-09-28 [VVoIPEP]

The following NIAP Technical Decisions (TD) apply to the [AppPP] and [VVoIPEP]. Their applicability to the evaluation was determined based on whether the TD is current (i.e. not superseded) and an SFR referenced by the TD was included in the ST:

TD	Applicable	PP/EP	Exclusion Rationale (if applicable)
TD0406 – FDP_IFF.1.5 Tests 1 and 2	Yes	VVoIPEP	
TD0392 – FCS_TLSC_EXT.1.2 Wildcard Checking	Yes	AppPP	
TD0390 – Cryptographically Secure RNG	No	AppPP	This TD applies to Windows platforms. The TOE runs on Android.
TD0389 – Handling of SSH EP claim for platform	Yes	AppPP	
TD0385 – FTP_DIT_EXT.1 Assurance Activity Clarification	No	AppPP	This TD only applies when conformance to the PP-Module for VPN Client Version 2.1 is also claimed.
TD0382 – Configuration Storage Options for Apps	No	AppPP	This TD applies to Windows and Linux platforms. The TOE runs on Android.
TD0380 – Linux Keyring Requirement in FCS_STO_EXT.1	Yes	AppPP	
TD0376 – Audit record entry for FMT_SMF.1 in FAU_GEN.1.2/VOIP	No	VVoIPEP	This TD is associated with FAU_GEN.1/VVOIP. The TOE does not include FAU_GEN.1/VVOIP functionality.
TD0372 – Auditing in VVOIP	Yes	VVoIPEP	

TD	Applicable	PP/EP	Exclusion Rationale (if applicable)
TD0368 – Audit Generation required in VVOIP	No	VVoIPEP	This TD was superseded by TD0372.
TD0367 – Trusted Updates	Yes	VVoIPEP	
TD0364 – Android mmap testing for FPT_AEX_EXT.1.1	Yes	AppPP	
TD0359 – Buffer Protection	Yes	AppPP	
TD0358 – Cipher Suites for TLS in SWApp v1.2	Yes	AppPP	
TD0327 – Default file permissions for FMT_CFG_EXT.1.2	Yes	AppPP	
TD0326 – RSA-based key establishment schemes	Yes	AppPP	
TD0305 – Handling of TLS connections with and without mutual authentication	Yes	AppPP	
TD0304 – Update to FCS_TLSC_EXT.1.2	Yes	AppPP	
TD0300 – Sensitive Data in FDP_DAR_EXT.1	Yes	AppPP	
TD0296 – Update to FCS_HTTPS_EXT.1.3	No	AppPP	This TD is associated with FCS_HTTPS_EXT.1. The TOE does not include FCS_HTTPS_EXT.1 functionality.
TD0295 – Update to FPT_AEX_EXT.1.3 Assurance Activities	Yes	AppPP	
TD0293 – Update to FCS_CKM.1(1)	No	AppPP	This TD was superseded by TD0326.
TD0283 – Cipher Suites for TLS in SWApp v1.2	No	AppPP	This TD was superseded by TD0358.
TD0279 – Ciphersuites for SRTP	Yes	VVoIPEP	
TD0269 – Update to FPT_AEX_EXT.1.3 Assurance Activity	No	AppPP	This TD was superseded by TD0295.
TD0268 – FMT_MEC_EXT.1 Clarification	Yes	AppPP	
TD0267 – TLSS testing - Empty Certificate Authorities list	No	AppPP	This TD is associated with FCS_TLSS_EXT.1. The TOE does not include FCS_TLSS_EXT.1 functionality.
TD0244 – FCS_TLSC_EXT - TLS Client Curves Allowed	Yes	AppPP	
TD0241 – Removal of Test 4.1 in FCS_TLSS_EXT.1.1	No	AppPP	This TD is associated with FCS_TLSS_EXT.1. The TOE does not include FCS_TLSS_EXT.1 functionality.

TD	Applicable	PP/EP	Exclusion Rationale (if applicable)
TD0238 – User-modifiable files FPT_AEX_EXT.1.4	Yes	AppPP	
TD0221 – FMT_SMF.1.1 - Assignments moved to Selections	Yes	AppPP	
TD0218 – Update to FPT_AEX_EXT.1.3 Assurance Activity	No	AppPP	This TD was superseded by TD0269.
TD0217 – Compliance to RFC5759 and RFC5280 for using CRLs	Yes	AppPP	
TD0215 – Update to FCS_HTTPS_EXT.1.2	Yes	AppPP	
TD0193 – Selection-Based FCS_COP.1 Added to VVoip EP to include AES-CTR Mode	Yes	VVoIPEP	
TD0192 – Update to FCS_STO_EXT.1 Application Note	No	AppPP	This TD was superseded by TD0380.
TD0178 – Integrity for installation tests in AppSW PP	Yes	AppPP	
TD0177 – FCS_TLSS_EXT.1 Application Note Update	Yes	AppPP	
TD0174 – Optional Ciphersuites for TLS	No	AppPP	TD0283 updates FCS_TLSC_EXT.1.1 in an incompatible manner, so both TDs cannot be applied.
TD0172 – Additional APIs added to FCS_RBG_EXT.1.1	No	AppPP	This TD was superseded by TD0390.
TD0163 – Update to FCS_TLSC_EXT.1.1 Test 5.4 and FCS_TLSS_EXT.1.1 Test	Yes	AppPP	
TD0142 – FAU_STG_EXT.1 - Optional Requirement	No	VVoIPEP	This TD was superseded by TD0368 and TD0372.
TD0131 – Update to FCS_TLSS_EXT.1.1 Test 4.5	No	AppPP	This TD is associated with FCS_TLSS_EXT.1. The TOE does not include FCS_TLSS_EXT.1 functionality.
TD0122 – FMT_SMF.1.1 Assignments moved to Selections	No	AppPP	This TD was superseded by TD0221.
TD0121 – FMT_MEC_EXT.1.1 Configuration Options	No	AppPP	This TD only applies when conformance to the Extended Package for Software File Encryption Version 1.0 is also claimed.
TD0119 – FCS_STO_EXT.1.1 in PP_APP_v1.2	Yes	AppPP	
TD0107 – FCS_CKM - ANSI X9.31-1998, Section 4.1.for Cryptographic Key Generation	Yes	AppPP	

TD	Applicable	PP/EP	Exclusion Rationale (if applicable)
TD0068 – Addition of SRTP Ciphersuites	No	VVoIPEP	TD0279 updates FCS_SRTP_EXT.1.2 in an incompatible manner, so both TDs cannot be applied.

Table 1: Technical Decisions

2.3 Conformance Rationale

This security target claims exact conformance to Protection Profile for Application Software Version 1.2 and Extended Package for Voice and Video over IP (VVoIP) Version 1.0.

The security problem definition, security objectives and security requirements in this Security Target are all taken from the Protection Profile and Extended Package performing only operations defined there.

3 Security Problem Definition

This section describes the assumptions and threats that are relevant to both the TOE and its environment.

The security problem definition has been taken from [AppPP] and [VVoIPEP] and is reproduced for the convenience of the reader.

3.1 Threats Addressed by the TOE

Threat	Description
T.NETWORK_ACCESS	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_EAVSDROP	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.
T.UNDETECTED_TRANSMISSION	An attacker may cause the TOE to exfiltrate audio and/or video media over a remote channel while in a state where the user has a reasonable expectation that no media is being transmitted.
T.CLOCK_DESYNC	An attacker may cause the TOE to use incorrect clock data, resulting in a denial of service from causing encryption and/or authentication connection failures.
T.MEDIA_DISCLOSURE	An attacker can use the encrypted variable rate vocoder frames to their advantage to decode transmitted data.

Table 2: Threats Addressed by the TOE

3.2 Organizational Security Policies

The [AppPP] and [VVoIPEP] do not specify any organizational security policies.

3.3 Assumptions

The following conditions are assumed to exist in the operational environment.

Assumption	Description
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: Assumptions on the Operational Environment

4 Security Objectives

This section defines the security objectives for the TOE and its supporting environment. The security objectives are intended to counter identified threats, comply with defined organizational security policies, and address applicable assumptions.

The security objectives are taken from [AppPP] and [VVoIPEP] and are reproduced for the convenience of the reader.

4.1 Security Objectives for the TOE

Security Objective	Description
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.
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.
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.
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.

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.

Table 4: Security Objectives for the TOE

4.2 Security Objectives for the Operational Environment

Security Objective	Description
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: Security Objectives for the Operational 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.

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 performed in the ST:

- Assignment: Indicated with ***bold italics*** text;
- Selection: Indicated with <u>underlined</u> text;
- Assignment within a Selection: Indicated with <u>***underlined bold italics***</u> text.

Note: The font conventions for Assignments and Assignments within a Selection overlap with existing [AppPP] or [VVoIPEP] formatting.

The ST does not perform any refinement or iteration operations.

Extended SFR are identified by having a label 'EXT' after the requirement name. Formatting conventions for operations performed by the [AppPP] or [VVoIPEP] are carried forward without modification.

5.2 TOE Security Functional Requirements

This section identifies the Security Functional Requirements for the TOE. The TOE Security Functional Requirements that appear below in Table 6 are described in more detail in the following subsections.

Requirement Class	Requirement Component
FCS: Cryptographic Support	FCS_CKM_EXT.1: Cryptographic Key Generation Services
	FCS_CKM.1(1): Cryptographic Asymmetric Key Generation
	FCS_CKM.2: Cryptographic Key Establishment
	FCS_COP.1(1): Cryptographic Operation - Encryption/Decryption
	FCS_COP.1(2): Cryptographic Operation - Hashing
	FCS_COP.1(3): Cryptographic Operation - Signing
	FCS_COP.1(4): Cryptographic Operation - Keyed-Hash Message Authentication
	FCS_COP.1(5): Cryptographic Operation - Encryption/Decryption for SRTP
	FCS_RBG_EXT.1: Random Bit Generation
	FCS_SRTP_EXT.1: Secure Real-Time Transport Protocol
	FCS_STO_EXT.1: Storage of Credentials
	FCS_TLSC_EXT.1: TLS Client Protocol

Requirement Class	Requirement Component
	FCS_TLSC_EXT.2: TLS Client Protocol
	FCS_TLSC_EXT.3: TLS Client Protocol
	FCS_TLSC_EXT.4: TLS Client Protocol
FCO: Communications	FCO_VOC_EXT.1: Fixed-Rate Vocoder
FDP: User Data Protection	FDP_DEC_EXT.1: Access to Platform Resources
	FDP_IFC.1: Subset Information Flow Control
	FDP_IFF.1: Information Flow Control Functions
	FDP_NET_EXT.1: Network Communications
	FDP_DAR_EXT.1: Encryption Of Sensitive Application Data
FIA: Identification and Authentication	FIA_X509_EXT.1: X.509 Certificate Validation
	FIA_X509_EXT.2: X.509 Certificate Authentication
FMT: Security Management	FMT_MEC_EXT.1: Supported Configuration Mechanism
	FMT_CFG_EXT.1: Secure by Default Configuration
	FMT_SMF.1: Specification of Management Functions
FPR: Privacy	FPR_ANO_EXT.1: User Consent for Transmission of Personally Identifiable Information
FPT: Protection of the TSF	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
FTA: TOE Access	FTA_SSL.3/Media: TSF-Initiated Termination (Media Channel)
FTP: Trusted Path/Channel	FTP_DIT_EXT.1: Protection of Data in Transit
	FTP_ITC.1/Control: Inter-TSF Trusted Channel (Signalling Channel)
	FTP_ITC.1/Media: Inter-TSF Trusted Channel (Media Channel)

Table 6: Security Functional Requirements

5.2.1 Cryptographic Support (FCS)

5.2.1.1 FCS_CKM_EXT.1 Cryptographic Key Generation Services

FCS_CKM_EXT.1.1 The application shall [*implement asymmetric cryptographic key generation*].

5.2.1.2 FCS_CKM.1(1) Cryptographic Asymmetric Key Generation

FCS_CKM.1.1(1) The application shall [implement functionality] to generate asymmetric cryptographic keys in accordance with a specified cryptographic key generation algorithm [

> ***[ECC schemes]** using **[“NIST curves” P-256, P-384 and [P-521]]** that meet the following: **[FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Appendix B.4]**,*

__[FFC schemes]__ using cryptographic key sizes of __[2048-bit or greater]__ that meet the following: __[FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.1]__

].

5.2.1.3 FCS_CKM.2 Cryptographic Key Establishment

FCS_CKM.2.1 The application shall [*implement functionality*] to perform cryptographic key establishment in accordance with a specified cryptographic key establishment method:

[

__[Elliptic curve-based key establishment schemes]__ that meets the following: __[NIST Special Publication 800-56A, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography"],__

__[Finite field-based key establishment schemes]__ that meets the following: __[NIST Special Publication 800-56A, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography"]__

].

5.2.1.4 FCS_COP.1(1) Cryptographic Operation - Encryption/Decryption

FCS_COP.1.1(1) The application shall perform encryption/decryption in accordance with a specified cryptographic algorithm

- AES-CBC (as defined in NIST SP 800-38A) mode;

and [

- *AES-GCM (as defined in NIST SP 800-38D)*

] and cryptographic key sizes 256-bit and [*128-bit*].

5.2.1.5 FCS_COP.1(2) Cryptographic Operation – Hashing

FCS_COP.1(2) The application shall perform cryptographic hashing services in accordance with a specified cryptographic algorithms [*SHA-1, SHA-256, SHA-384, SHA-512*] and message digest sizes [*160, 256, 384, 512*] bits that meet the following: FIPS Pub 180-4.

5.2.1.6 FCS_COP.1(3) Cryptographic Operation – Signing

FCS_COP.1.1(3) The application shall perform cryptographic signature services (generation and verification) in accordance with a specified cryptographic algorithm [

__RSA schemes__ using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 4,

__ECDSA schemes__ using "NIST curves" P-256, P-384 and [P-521] that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 5

].

5.2.1.7 FCS_COP.1(4) Cryptographic Operation - Keyed-Hash Message Authentication

FCS_COP.1.1(4) The application shall perform keyed-hash message authentication in accordance with a specified cryptographic algorithm

- HMAC-SHA-256

and [

- *SHA-1,*
- *SHA-384*

] with key sizes [***160-bits, 256-bits, 384-bits***] and message digest sizes 256 and [*160, 384*] bits that meet the following: FIPS Pub 198-1 *The Keyed-Hash Message Authentication Code* and FIPS Pub 180-4 *Secure Hash Standard*.

5.2.1.8 FCS_COP.1(5) Cryptographic Operation - Encryption/Decryption for SRTP

FCS_COP.1.1(5) Refinement: The application shall perform encryption/decryption **to support SDES-SRTP** in accordance with a specified cryptographic algorithm

- AES-CTR (as defined in NIST SP 800-38A) mode;

and [

- *AES-GCM (as defined in NIST SP 800-38D)*

] and cryptographic key sizes 128-bit and [*256-bit*].

5.2.1.9 FCS_RBG_EXT.1 Random Bit Generation Services

FCS_RBG_EXT.1.1 The application shall [*invoke platform-provided DRBG functionality*] for its cryptographic operations.

5.2.1.10 FCS_SRTP_EXT.1 Secure Real-Time Transport Protocol

FCS_SRTP_EXT.1.1 The TSF shall implement the Secure Real-Time Transport Protocol (SRTP) that complies with RFC 3711, and use Security Descriptions for Media Streams (SDES) in compliance with RFC 4568 to provide key information for the SRTP connection.

FCS_SRTP_EXT.1.2 The TSF shall implement SDES-SRTP supporting the following ciphersuites [

- AES_CM_128_HMAC_SHA1_80, in accordance with RFC 4568,
- AES_CM_128_HMAC_SHA1_32, in accordance with RFC 4568,
- AES_256_CM_HMAC_SHA1_80, in accordance with RFC 6188,
- AES_256_CM_HMAC_SHA1_32, in accordance with RFC 6188,
- AEAD_AES_128_GCM, in accordance with RFC7714,
- AEAD_AES_256_GCM, in accordance with RFC 7714].

FCS_SRTP_EXT.1.3 The TSF shall ensure the SRTP NULL algorithm can be disabled.

FCS_SRTP_EXT.1.4 The TSF shall allow the SRTP ports to be used for SRTP communications to be specified by an Authorized Administrator.

5.2.1.11 FCS_STO_EXT.1 Storage of Credentials

FCS_STO_EXT.1.1 The application shall [

invoke the functionality provided by the platform to securely store [***Key Encrypting Key***],

implement functionality to securely store [***X.509 certificates and associated private keys***]

] to non-volatile memory.

5.2.1.12 FCS_TLSC_EXT.1 TLS Client Protocol

FCS_TLSC_EXT.1.1 The application shall [*implement TLS 1.2 (RFC 5246)*] 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_ SHA256 as defined in RFC 5246
- TLS_DHE_RSA_WITH_AES_256_CBC_ SHA256 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_GCM_SHA256 as defined in RFC 5289
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289

].

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.

5.2.1.13 FCS_TLSC_EXT.2 TLS Client Protocol

FCS_TLSC_EXT.2.1 The application shall support mutual authentication using X.509v3 certificates.

5.2.1.14 FCS_TLSC_EXT.3 TLS Client Protocol

FCS_TLSC_EXT.3.1 The application shall present the signature_algorithms extension in the Client Hello with the supported_signature_algorithms value containing the following hash algorithms: [*SHA256, SHA384, SHA512*] and no other hash algorithms.

5.2.1.15 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*].

5.2.2 Communications (FCO)

5.2.2.1 FCO_VOC_EXT.1 Fixed-Rate Vocoder

FCO_VOC_EXT.1 The TSF shall transmit voice media using a constant bit rate vocoder.

5.2.3 User Data Protection (FDP)

5.2.3.1 FDP_DEC_EXT.1 Access to Platform Resources

FDP_DEC_EXT.1.1 The application shall restrict its access to [

network connectivity,

microphone,

].

FDP_DEC_EXT.1.2 The application shall restrict is access to [

Android keystore

].

5.2.3.2 FDP_IFC.1 Subset Information Flow Control

FDP_IFC.1.1 The TSF shall enforce the [*media transmission policy*] on [*voice/video media transmitted by the TOE*].

5.2.3.3 FDP_IFF.1 Information Flow Control Functions

FDP_IFF.1.1 The TSF shall enforce the [*media transmission policy*] based on the following types of subject and information security attributes: [*ESC registration status and TOE hook state*].

FDP_IFF.1.2 The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold: [

- *The TOE is registered with the ESC,*
- *A call has been established with a telephony device (VVoIP endpoint),*
- *The TOE is in the off-hook state,*
- *The TOE is not in the mute state,*
- ***[No other rules]***].

FDP_IFF.1.3 The TSF shall enforce the [*no additional information flow control policy rules*].

FDP_IFF.1.4 The TSF shall explicitly authorize an information flow based on the following rules: [*no additional rules*].

FDP_IFF.1.5 The TSF shall explicitly deny an information flow based on the following rules: [*all TCP and UDP ports used by the TOE are closed when not in active use*].

5.2.3.4 FDP_NET_EXT.1 Network Communications

FDP_NET_EXT.1.1 The application shall restrict network communications to [

user-initiated communication for [a SIP server, a VVoIP endpoint, check for updates],

[certificate validation using CRL, fetch timeout configuration from the configuration server]

].

5.2.3.5 FDP_DAR_EXT.1 Encryption Of Sensitive Application Data

FDP_DAR_EXT.1.1 The application shall [

leverage platform-provided functionality to encrypt sensitive data

] in non-volatile memory.

5.2.4 Identification and Authentication (FIA)

5.2.4.1 FIA_X509_EXT.1 X.509 Certificate Validation

FIA_X509_EXT.1.1 The application shall [*implement 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 [*a Certificate Revocation List (CRL) as specified in RFC 5280 Section 6.3*].
- 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 theextendedKeyUsage field.

FIA_X509_EXT.1.2 The application shall treat a certificate as a CA certificate only if the basicConstraints Extension is present and the CA flag is set to TRUE.

5.2.4.2 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*].

5.2.5 Security Management (FMT)

5.2.5.1 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.

5.2.5.2 *FMT_CFG_EXT.1 Secure by Default Configuration*

FMT_CFG_EXT.1.1 The application shall provide only 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 the application's binaries and data files from modification by normal unprivileged user.

5.2.5.3 *FMT_SMF.1 Specification of Management Functions*

FMT_SMF.1.1 The TSF shall be capable of performing the following management functions:

[

- ***configure the termination period for idle calls,***
- [***Configure SIP server,***
- ***Load X.509 Certificates and private keys,***
- ***Select X.509 Certificate,***
- ***Configure Update check server,***
- ***Configure Configuration server***]

].

5.2.6 Privacy (FPR)

5.2.6.1 *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*].

5.2.7 Protection of the TSF (FPT)

5.2.7.1 *FPT_API_EXT.1 Use of Supported Services and APIs*

FPT_API_EXT.1.1 The application shall only use documented platform APIs.

5.2.7.2 *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.

5.2.7.3 FPT_TUD_EXT.1 Integrity for Installation and Update

FPT_TUD_EXT.1.1 The application shall [*provide the ability*] 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 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.

5.2.7.4 FPT_LIB_EXT.1 Use of Third Party Libraries

FPT_LIB_EXT.1.1 The application shall be packaged with only [***PJSIP, OpenSSL (FIPS build), libSRTP, Open H.264, SQLCipher, Butterknife, GlowPadBackport, Eventbus, Crouton, Kotlin, Kotlin (Coroutines), Volley, Gson***].

5.2.8 TOE Access (FTA)

5.2.8.1 FTA_SSL.3/Media TSF-Initiated Termination (Media Channel)

FTA_SSL.3.1/Media The TSF shall terminate **voice/video transmission** after [*[**30**]* *seconds, an administrator configurable interval on the* [configuration server] *downloaded to the TOE during configuration*].

5.2.9 Trusted Path/Channel (FTP)

5.2.9.1 FTP_DIT_EXT.1 Protection of Data in Transit

FTP_DIT_EXT.1.1 The application shall [*encrypt all transmitted sensitive data with [TLS, **SRTP**]*] between itself and another trusted IT product.

5.2.9.2 FTP_ITC.1/Control Inter-TSF Trusted Channel (Signaling Channel)

FTP_ITC.1.1/Control The TSF shall **be capable of using [*SIP*]** to provide a trusted communication channel between itself and **an Enterprise Session Controller** that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from disclosure and detection of modification of the channel data.

FTP_ITC.1.2/Control The TSF shall permit **[*the TSF, the Enterprise Session Controller*]** to initiate communication via the trusted channel.

FTP_ITC.1.3/Control The TSF shall initiate communication via the trusted channel for [*establishment of call control*].

5.2.9.3 *FTP_ITC.1/Media Inter-TSF Trusted Channel (Media Channel)*

FTP_ITC.1.1/Media The TSF shall **be capable of using [*SRTP*]** to provide a trusted communication channel between itself and **another VVoIP endpoint or other telephony device** that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from disclosure and detection of modification of the channel data.

FTP_ITC.1.2/Media The TSF shall permit **[*the TSF, another VVoIP endpoint or other telephony device*]** to initiate communication via the trusted channel.

FTP_ITC.1.3/Media The TSF shall initiate communication via the trusted channel for [*transmission of voice/video media*].

5.3 Security Assurance Requirements

All assurance requirements are summarized in the table below.

Requirement Class	Requirement Component
ASE: Security Target	ASE_INT.1: ST Introduction ASE_CCL.1: Conformance Claims ASE_OBJ.1: Security Objectives for the Operational Environment ASE_ECD.1: Extended Components Definition ASE_REQ.1: Stated Security Requirements ASE_TSS.1: TOE Summary Specification
ADV: Development	ADV_FSP.1: Basic Functional Specification
AGD: Guidance Documents	AGD_OPE.1: Operational User Guidance
	AGD_PRE.1: Preparative User Guidance
ALC: Life-cycle Support	ALC_CMC.1: Labeling of the TOE
	ALC_CMS.1: TOE CM Coverage
	ALC_TSU_EXT.1: Timely Security Updates
ATE: Tests	ATE_IND.1: Independent Testing
AVA: Vulnerability Assessment	AVA_VAN.1: Vulnerability Survey

Table 7: Security Assurance Requirements

5.3.1 Class ASE: Security Target

As per the activities divined in [CEM].

5.3.2 Class ADV: Development

5.3.2.1 *ADV_FSP.1 Basic Functional Specification*

Developer action elements:

ADV_FSP.1.1D	The developer shall provide a functional specification.
ADV_FSP.1.2D	The developer shall provide a tracing from the functional specification to the SFRs.

Content and presentation elements:

ADV_FSP.1.1C	The functional specification shall describe the purpose and method of use for each SFR-enforcing and SFR-supporting TSFI.
ADV_FSP.1.2C	The functional specification shall identify all parameters associated with each SFR-enforcing and SFR-supporting TSFI.
ADV_FSP.1.3C	The functional specification shall provide rationale for the implicit categorization of interfaces as SFR-non-interfering.
ADV_FSP.1.4C	The tracing shall demonstrate that the SFRs trace to TSFIs in the functional specification

Evaluator action elements:

ADV_FSP.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
ADV_FSP.1.2E	The evaluator shall determine that the functional specification is an accurate and complete instantiation of the SFRs.

5.3.3 Class AGD: Guidance Documents

The guidance documents will be provided with the ST. Guidance must include a description of how the IT personnel verifies that the Operational Environment can fulfil its role for the security functionality. The documentation should be in an informal style and readable by the IT personnel. Guidance must be provided for every operational environment that the product supports as claimed in the ST. This guidance includes instructions to successfully install the TSF in that environment; and Instructions to manage the security of the TSF as a product and as a component of the larger operational environment. Guidance pertaining to particular security functionality is also provided; requirements on such guidance are contained in the assurance activities specified with each requirement.

5.3.3.1 AGD_OPE.1 Operational User Guidance

Developer action elements:

AGD_OPE.1.1D	The developer shall provide operational user guidance.

Content and presentation elements:

AGD_OPE.1.1C	The operational user guidance shall describe, for each user role, the user-accessible functions and privileges that should be controlled in a secure processing environment, including appropriate warnings.
AGD_OPE.1.2C	The operational user guidance shall describe, for each user role, how to use the available interfaces provided by the TOE in a secure manner.
AGD_OPE.1.3C	The operational user guidance shall describe, for each user role, the available functions and interfaces, in particular all security parameters under the control of the user, indicating secure values as appropriate.
AGD_OPE.1.4C	The operational user guidance shall, for each user role, clearly present each type of security-relevant event relative to the user-accessible

	functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.
AGD_OPE.1.5C	The operational user guidance shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences, and implications for maintaining secure operation.
AGD_OPE.1.6C	The operational user guidance shall, for each user role, describe the security measures to be followed in order to fulfil the security objectives for the operational environment as described in the ST.
AGD_OPE.1.7C	The operational user guidance shall be clear and reasonable.

Evaluator action elements:

AGD_OPE.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence. **Assurance Activity:** Some of the contents of the operational guidance will be verified by the assurance activities in [AppPP] Section 5.1 and evaluation of the TOE according to the [CEM]. The following additional information is also required. If cryptographic functions are provided by the TOE, the operational guidance shall contain instructions for configuring the cryptographic engine associated with the evaluated configuration of the TOE. It shall provide a warning to the administrator that use of other cryptographic engines was not evaluated nor tested during the CC evaluation of the TOE. The documentation must describe the process for verifying updates to the TOE by verifying a digital signature – this may be done by the TOE or the underlying platform. The evaluator shall verify that this process includes the following steps: Instructions for obtaining the update itself. This should include instructions for making the update accessible to the TOE (e.g., placement in a specific directory). Instructions for initiating the update process, as well as discerning whether the process was successful or unsuccessful. This includes generation of the hash/digital signature. The TOE will likely contain security functionality that does not fall in the scope of evaluation under the [AppPP]. The operational guidance shall make it clear to an administrator which security functionality is covered by the evaluation activities.

5.3.3.2 AGD_PRE.1 Preparative Procedures

Developer action elements:

AGD_PRE.1.1D	The developer shall provide the TOE, including its preparative procedures.

Content and presentation elements:

AGD_PRE.1.1C	The preparative procedures shall describe all the steps necessary for secure acceptance of the delivered TOE in accordance with the developer's delivery procedures.
AGD_PRE.1.2C	The preparative procedures shall describe all the steps necessary for secure installation of the TOE and for the secure preparation of the operational environment in accordance with the security objectives for the operational environment as described in the ST.

Evaluator action elements:

AGD_PRE.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
AGD_PRE.1.2E	The evaluator shall apply the preparative procedures to confirm that the TOE can be prepared securely for operation. **Assurance Activity:** As indicated in the introduction above, there are significant expectations with respect to the documentation—especially when configuring the operational environment to support TOE functional requirements. The evaluator shall check to ensure that the guidance provided for the TOE adequately addresses all platforms claimed for the TOE in the ST.

5.3.4 Class ALC: Life-cycle Support

5.3.4.1 ALC_CMC.1 Labeling of the TOE

Developer action elements:

ALC_CMC.1.1D	The developer shall provide the TOE and a reference for the TOE.

Content and presentation elements:

ALC_CMC.1.1C	The TOE shall be labeled with a unique reference.

Evaluator action elements:

ALC_CMC.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence. **Assurance Activity:** The evaluator shall check the ST to ensure that it contains an identifier (such as a product name/version number) that specifically identifies the version that meets the requirements of the ST. Further, the evaluator shall check the AGD guidance and TOE samples received for testing to ensure that the version number is consistent with that in the ST. If the vendor maintains a web site advertising the TOE, the evaluator shall

	examine the information on the web site to ensure that the information in the ST is sufficient to distinguish the product.

5.3.4.2 ALC_CMS.1 TOE CM Coverage

Developer action elements:

ALC_CMS.1.1D	The developer shall provide a configuration list for the TOE.

Content and presentation elements:

ALC_CMS.1.1C	The configuration list shall include the following: the TOE itself; and the evaluation evidence required by the SARs.
ALC_CMS.1.2C	The configuration list shall uniquely identify the configuration items.

Evaluator action elements:

ALC_CMS.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence. **Assurance Activity:** The “evaluation evidence required by the SARs” in this PP is limited to the information in the ST coupled with the guidance provided to administrators and users under the AGD requirements. By ensuring that the TOE is specifically identified and that this identification is consistent in the ST and in the AGD guidance (as done in the assurance activity for ALC_CMC.1), the evaluator implicitly confirms the information required by this component. Life-cycle support is targeted aspects of the developer’s life-cycle and instructions to providers of applications for the developer’s devices, rather than an in-depth examination of the TSF manufacturer’s development and configuration management process. This is not meant to diminish the critical role that a developer’s practices play in contributing to the overall trustworthiness of a product; rather, it’s a reflection on the information to be made available for evaluation. The evaluator shall ensure that the developer has identified (in guidance documentation for application developers concerning the targeted platform) one or more development environments appropriate for use in developing applications for the developer’s platform. For each of these development environments, the developer shall provide information on how to configure the environment to ensure that buffer overflow protection mechanisms in the environment(s) are invoked (e.g., compiler flags). The evaluator shall ensure that this documentation also includes an indication of whether such protections are on by default, or have to be specifically enabled. The evaluator shall ensure that the TSF is uniquely identified (with respect to other products from the TSF vendor), and that documentation provided by the developer in association with the

	requirements in the ST is associated with the TSF using this unique identification.

5.3.4.3 ALC_TSU_EXT.1 Timely Security Updates

Developer action elements:

ALC_TSU_EXT.1.1D	The developer shall provide a description in the TSS of how timely security updates are made to the TOE.
ALC_TSU_EXT.1.2D	The developer shall provide a description in the TSS of how users are notified when updates change security properties or the configuration of the product.

Content and presentation elements:

ALC_TSU_EXT.1.1C	The description shall include the process for creating and deploying security updates for the TOE software.
ALC_TSU_EXT.1.2C	The description shall express the time window as the length of time, in days, between public disclosure of a vulnerability and the public availability of security updates to the TOE.
ALC_TSU_EXT.1.3C	The description shall include the mechanisms publicly available for reporting security issues pertaining to the TOE.

Evaluator action elements:

ALC_TSU_EXT.1.1E	The evaluator *shall confirm* that the information provided meets all requirements for content and presentation of evidence. **Assurance Activity:** The evaluator shall verify that the TSS contains a description of the timely security update process used by the developer to create and deploy security updates. The evaluator shall verify that this description addresses the entire application. The evaluator shall also verify that, in addition to the TOE developer's process, any third-party processes are also addressed in the description. The evaluator shall also verify that each mechanism for deployment of security updates is described. The evaluator shall verify that, for each deployment mechanism described for the update process, the TSS lists a time between public disclosure of a vulnerability and public availability of the security update to the TOE patching this vulnerability, to include any third-party or carrier delays in deployment. The evaluator shall verify that this time is expressed in a number or range of days. The evaluator shall verify that this description includes the publicly available mechanisms (including either an email address or website) for reporting security issues related to the TOE. The evaluator shall verify that the description of this mechanism includes a method for protecting

	the report either using a public key for encrypting email or a trusted channel for a website.

5.3.5 Class ATE: Tests

5.3.5.1 *ATE_IND.1 Independent Testing – Conformance*

Developer action elements:

ATE_IND.1.1D	The developer shall provide the TOE for testing.

Content and presentation elements:

ATE_IND.1.1C	The TOE shall be suitable for testing.

Evaluator action elements:

ATE_IND.1.1E	The evaluator *shall confirm* that the information provided meets all requirements for content and presentation of evidence.
ATE_IND.1.2E	The evaluator shall test a subset of the TSF to confirm that the TSF operates as specified. **Assurance Activity:** The evaluator shall prepare a test plan and report documenting the testing aspects of the system, including any application crashes during testing. The evaluator shall determine the root cause of any application crashes and include that information in the report. The test plan covers all of the testing actions contained in the [CEM] and the body of the [AppPP]'s Assurance Activities. While it is not necessary to have one test case per test listed in an Assurance Activity, the evaluator must document in the test plan that each applicable testing requirement in the ST is covered. The test plan identifies the platforms to be tested, and for those platforms not included in the test plan but included in the ST, the test plan provides a justification for not testing the platforms. This justification must address the differences between the tested platforms and the untested platforms, and make an argument that the differences do not affect the testing to be performed. It is not sufficient to merely assert that the differences have no affect; rationale must be provided. If all platforms claimed in the ST are tested, then no rationale is necessary. The test plan describes the composition of each platform to be tested, and any setup that is necessary beyond what is contained in the AGD documentation. It should be noted that the evaluator is expected to follow the AGD documentation for installation and setup of each platform either as part of a test or as a standard pre-test condition. This may include special test drivers or tools. For each driver or tool, an argument (not just an assertion) should be provided that the driver or

	tool will not adversely affect the performance of the functionality by the TOE and its platform. This also includes the configuration of the cryptographic engine to be used. The cryptographic algorithms implemented by this engine are those specified by this PP and used by the cryptographic protocols being evaluated (IPsec, TLS, SSH). The test plan identifies high-level test objectives as well as the test procedures to be followed to achieve those objectives. These procedures include expected results. The test report (which could just be an annotated version of the test plan) details the activities that took place when the test procedures were executed, and includes the actual results of the tests. This shall be a cumulative account, so if there was a test run that resulted in a failure; a fix installed; and then a successful re-run of the test, the report would show a “fail” and “pass” result (and the supporting details), and not just the “pass” result.

5.3.6 Class AVA: Vulnerability Assessment

5.3.6.1 AVA VAN.1 Vulnerability Survey

Developer action elements:

AVA_VAN.1.1D	The developer shall provide the TOE for testing.

Content and presentation elements:

AVA_VAN.1.1C	The TOE shall be suitable for testing.

Evaluator action elements:

AVA_VAN.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
AVA_VAN.1.2E	The evaluator shall perform a search of public domain sources to identify potential vulnerabilities in the TOE.
AVA_VAN.1.3E	The evaluator shall conduct penetration testing, based on the identified potential vulnerabilities, to determine that the TOE is resistant to attacks performed by an attacker possessing Basic attack potential. **Assurance Activity:** The evaluator shall generate a report to document their findings with respect to this requirement. This report could physically be part of the overall test report mentioned in ATE_IND, or a separate document. The evaluator performs a search of public information to find vulnerabilities that have been found in similar applications with a particular focus on network protocols the application uses and document formats it parses. The evaluator shall also run a virus scanner with the most current virus

	definitions against the application files and verify that no files are flagged as malicious. The evaluator documents the sources consulted and the vulnerabilities found in the report. For each vulnerability found, the evaluator either provides a rationale with respect to its non-applicability, or the evaluator formulates a test (using the guidelines provided in ATE_IND) to confirm the vulnerability, if suitable. Suitability is determined by assessing the attack vector needed to take advantage of the vulnerability. If exploiting the vulnerability requires expert skills and an electron microscope, for instance, then a test would not be suitable and an appropriate justification would be formulated.

6 TOE SUMMARY SPECIFICATION

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

Requirement	Rationale
FCS_CKM_EXT.1 FCS_CKM.1(1) FCS_CKM.2	The TOE generates asymmetric elliptic curve cryptographic keys to perform elliptic curve-based key establishment when ECDHE ciphersuites are negotiated. The elliptic curve-based key establishment is complaint with NIST SP 800-56Ar3 and supports P-256, P-384, and P-521. ECDSA Cert. #C 577 KAS-ECC-Component Cert. #C 577 The TOE generates asymmetric finite field cryptographic keys to perform finite field-based key establishment when DHE ciphersuites are negotiated. The finite field-based key establishment is complaint with NIST SP 800-56Ar3 and supports Group 14. CAVP testing is not available for Group 14 and finite field-based key establishment using safe-primes. DSA Cert. #C 577 KAS-FFC-Component Cert. #C 577 The TOE implements the above key generation and key establishment in the Cellcrypt Classified V2.0 cryptographic library version 2.9. This library is based on the OpenSSL (FIPS build) v2.0.10 library.

FCS_COP.1(1)
FCS_COP.1(5)
	The TOE implements AES encryption and decryption in the Cellcrypt Classified V2.0 library. The TOE implements CBC, CTR, and GCM modes with 128-bit and 256-bit keys.
	AES Cert. #C 577

FCS_COP.1(2)	The TOE implements SHA hashing in the Cellcrypt Classified V2.0 library. SHA-1, SHA-256, SHA-384, SHA-512 are supported. SHS Cert. #C 577
FCS_COP.1(3)	The TOE implements RSA and ECDSA signature generation and verification in the Cellcrypt Classified V2.0 library. RSA keys of 2048-bits or greater are supported. ECDSA keys using P-256, P-384, and P-521 are supported. RSA Cert. #C 577 ECDSA Cert. #C 577
FCS_COP.1(4)	The TOE implements HMAC message authentication in the Cellcrypt Classified V2.0 library. HMAC-SHA-1, HMAC-SHA-256, and HMAC-SHA-384 are supported. HMAC Cert. #TC 577BD
FCS_RBG_EXT.1	The TOE invokes the platform provided DRBG for all random bit generation using the /dev/urandom call.

Requirement	Rationale
FCS_SRTP_EXT.1 FTP_ITC.1/Media	The TOE implements the Secure Real-Time Transport Protocol (SRTP) in the libSRTP v.1.5.4 library. libSRTP is compatible with SRTP (RFC 3711) and SRTP SDES (RFC 4568). libSRTP calls the Cellcrypt Classified V2.0 library to perform cryptographic operations. The TOE supports the following SRTP ciphersuites: • AES_CM_128_HMAC_SHA1_80, in accordance with RFC 4568, • AES_CM_128_HMAC_SHA1_32, in accordance with RFC 4568, • AES_256_CM_HMAC_SHA1_80, in accordance with RFC 6188, • AES_256_CM_HMAC_SHA1_32, in accordance with RFC 6188, • AEAD_AES_128_GCM, in accordance with RFC7714, • AEAD_AES_256_GCM, in accordance with RFC 7714 The TOE establishes SRTP sessions (for both incoming and outgoing calls) using SIP, described in FTP_ITC.1/Control. The SRTP keying material and ciphersuites are negotiated using SDES (SDP attachment to a SIP message). The TOE rejects the NULL ciphersuite as well as any other ciphersuite not listed above.
FCS_STO_EXT.1	The TOE invokes the Android Keystore API to store the AES 256-bit Key Encrypting Key (KEK) using the BouncyCastle Keystore provider. The following API is used: https://developer.android.com/reference/java/security/KeyStore.html The TOE uses the KEK and FCS_COP.1(1) encryption/decryption to securely store the X.509 certificates and associated private keys in the /data/data/com.cellcrypt.cellcryptclassified/filescellcrypt.BKS file. The MODE_PRIVATE flag is also set on this file.

Requirement	Rationale
FCS_TLSC_EXT.1 FCS_TLSC_EXT.2 FCS_TLSC_EXT.3 FCS_TLSC_EXT.4	The TOE implements TLS v1.2 as specified in RFC 5246 and supports the following ciphersuites: • 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_SHA256 as defined in RFC 5246 • TLS_DHE_RSA_WITH_AES_256_CBC_SHA256 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_GCM_SHA256 as defined in RFC 5289 • TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 The TOE establishes the reference identifier by parsing the DNS Name or IP address for the configured SIP server. The reference identifier is matched against the SAN, if present. If the SAN is not present, the referenced identifier is matched against the CN. The TOE does not support certificate pinning or wildcards in the DNS name of the server certificate. The TOE presents the supported Elliptic Curves Extension in the Client Hello message with the P-256, P-384, and P-521 curves. This is the default TOE behavior and cannot be modified. The TOE also presents the Signature Algorithms extension in the Client Hello message indicating support for SHA-256, SHA-384, and SHA-512 signature hashes. This is the default TOE behavior and cannot be modified.
FCO_VOC_EXT.1	The TOE uses the G.711 vocoder to transmit voice media. This vocoder generates generate a constant bit-rate stream of 64 kbit/s.
FDP_DEC_EXT.1	The TOE accesses network connectivity and microphone hardware resources. The TOE Android Keystore is the only sensitive information repository the TOE accesses.
FDP_IFC.1 FDP_IFF.1	The TOE does not transmit any media data when it is not on a call. The TOE does not transmit any media data when it is muted. The TOE does not implement a “hold” state.

Requirement	Rationale
FDP_NET_EXT.1	The TOE performs the following user-initiated network communications: • Communicating with to a SIP server. • Communicating with a VVoIP endpoint. • Communicating with an Update check server. The TOE automatically initiates the following network communications: • CRL certificate validation. • Inactive call timeout setting update from the configuration server.
FDP_DAR_EXT.1	All data (both sensitive and non-sensitive data) is stored in the /data/data/com.cellcrypt.cellcryptclassified/filescellcrypt.BKS file. The following sensitive data is stored in this file: • SIP_NUMBERs • SIP_IDs The file is protected using the platform MODE_PRIVATE flag as well as AES-256-CBC encryption. The encryption is performed according to FCS_STO_EXT.1 since credentials are also stored in this file.
FIA_X509_EXT.1 FIA_X509_EXT.2	The TSF uses X.509v3 uses certificates to authenticate the user to the SIP server via a mutually authenticated TLS connection. The TOE performs validity checks on the CA path and that either the SubjectAltName or CN match what was provided on the distant connection certificate. The TSF also performs CRL validity checks on the certificate. Connection attempts are made only if the certificate is deemed valid. The TSF performs checks for RFC 5280 validation, validates certificate path by ensuring the basicConstraints extension is present and the CA flag is set to True for all CA certificates. The OE also verifies the path terminates with a trust anchor that was manually imported into the TOE. The TSF validates the extendedKeyUsage field according to the following rules: • 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. In the case where it is not possible to check the validity of the Certificate via an online check the TOE does not establish a TLS connection.
FMT_MEC_EXT.1	The TOE stores its configuration setting using the method supported by the Android platform (i.e. an XML file in the /data/data/com.cellcrypt.cellcryptclassified/shared_prefs directory).
FMT_CFG_EXT.1	The TOE does not contain any credential when it is installed. Certificates and passwords must be configured by the user before the TOE is operational.

Requirement	Rationale
FMT_SMF.1	The TOE has a GUI which allows users to specify a SIP server address, import an X.509 certificate, select the X.509 certificate used to authenticate a TLS connection, Update check server address, and a Configuration server address. The allows the configuration of the termination period for idle calls through a setting on the configuration server.
FPR_ANO_EXT.1	The TOE does not transmit PII.
FPT_API_EXT.1	The TOE uses the following Android APIs: android, android.annotation, android.app, android.bluetooth, android.content, android.content.res, android.database, android.database.sqlite, android.graphics, android.hardware, android.media, android.media.audiofx, android.net, android.os, android.preference, android.provider, android.security, android.telephony, android.test, android.text, android.text.method, android.util, android.view, android.view.inputmethod, android.widget, java.io, java.lang, java.lang.annotation, java.lang.ref, java.lang.reflect, java.math, java.net, java.nio, java.nio.channels, java.nio.charset, java.security, java.security.cert, java.security.interfaces, java.text, java.util, java.util.concurrent, java.util.concurrent.atomic, java.util.concurrent.atomic.locks, java.util.regex, javax.crypto, javax.crypto.spec, and org.xml.
FPT_AEX_EXT.1	The TOE is compiled with the -fPIC and -fpic compiler flags which enable ASLR. The TOE is compiled with the -fstack-protector-all compiler flag which enables stack-based buffer overflow protection.
FPT_TUD_EXT.1	The TOE checks for updates by querying the Update check server which returns the most recent version number. The TOE indicates an update is available when the returned version is greater than the TOE version. Updates to the TOE are distributed in the Android APK format. Cellcrypt is the authorized source of TOE updates. All TOE APK files are signed with the Cellcrypt private key. Upon initial installation, Android trusts the associated public key. Android uses the associated public key to verify the authenticity of all subsequent updates to the TOE software. The TOE allows the user to query the application's version in the user settings.

Requirement	Rationale
FPT_LIB_EXT.1	The TOE includes the following third-party libraries: • PJSIP • Cellcrypt Classified V2.0 cryptographic library (based on OpenSSL (FIPS build)) • libSRTP • Open H.264 • SQLCipher • Butterknife • GlowPadBackport • Eventbus • Crouton • Kotlin • Kotlin (Coroutines) • Volley • Gson
FTA_SSL.3/Media	The TOE terminates idle voice connections. The TOE considers voice connection idle when the TOE is not receiving data from the peer. The idle time is 30 seconds by default. The Administrator can update the idle time through a configuration file the TOE downloads from the configuration server.
FTP_DIT_EXT.1	The TOE encrypts all sensitive data using TLS or SRTP. TLS is used to encrypt the control channel described by FTP_ITC.1/Control while SRTP is used to encrypt the media channel described by FTP_ITC.1/Media.
FTP_ITC.1/Control	The TOE implements the Session Initiation Protocol (SIP) in the PJSIP v2.1 library. The PJSIP library calls the OpenSSL (FIPS build) v2.0.10 library to perform cryptographic operations. The TOE uses TLSv1.2 (FCS_TLSC_EXT) to protect the SIP communications.
ALC_TSU_EXT.1	The developer's process for providing timely security updates involves accepting reports about potential vulnerabilities on their webpage at https://www.csghq.com/about. The use of https protects the reports from unauthorized disclosure. Upon receipt of a report, the developer identifies remedial action. Once the remedial action has been implemented, the TOE undergoes normal production testing before being released. Each customer identified security officer is notified via email when a security update is available. The time between disclosure of a vulnerability and availability of a security update varies from two weeks to 90 days.

Table 8: TOE Summary Specification Description

APPENDIX A – TERMINOLOGY AND ACRONYMS

Term	Description
AES	Advanced Encryption Standard
CA	Certificate Authority
CAVP	Cryptographic Algorithm Validation Program
CC	Common Criteria for Information Technology Security Evaluation
CIK	Crypto Ignition Key
CM	Cellcrypt Mobile
DRBG	Deterministic Random Bit Generator
FIPS	Federal Information Processing Standard
KEK	Key Encryption Key
NAT	Network Address Translation
NIAP	National Information Assurance Partnership
NIST	National Institute of Standards and Technology
PKI	Public Key Infrastructure
PP	NIAP Protection Profiles
PRNG	Pseudo Random Number Generator
RNG	Random Number Generator
RTP	Real Time Protocol (RFC 3550)
SDES	SDP Security Descriptions for Media Streams (RFC 4568)
SDP	Session Description Protocol (RFC 4566)
SIP	Session Initiation Protocol (RFC 3261)
SRTP	Secure Real Time Protocol (RFC 3711)
STUN	Session Traversal Utilities for NAT
SVoIP	Secure Voice over Internet Protocol
TEE	ARM Trusted Execution Environment
TOE	Target Of Evaluation
TSF	TOE Security Functions
VPN	Virtual Private Network
VoIP	Voice over Internet Protocol

Table 9: Terminology and Acronyms