Cellcrypt®
TRUSTED COMMUNICATIONS

Security Target

Cellcrypt Android Mobile Client version 4.40

Ref:	ST-FED-MCL-And-1
Ver:	1.2.4
Date:	Sep 19, 2022

Contents

Figures

Tables

Releases

Issue	Description
0.1.0	Initial release.
0.2.1	Added support for the NIAP TLS Package
0.2.2	Included Samsung S7 with Android 7
0.3.0	Updated to replaced EP_VVoIP conformance to MOD_VVoIP conformance
0.4.0	Added Figure 1. Added new TD's. Updated doc according to evaluator feedback
0.4.1	Updated with latest TD's. Answered evaluator questions.
0.5.0	Updated according to evaluator feedback 6 Aug 2021
0.5.1	Updated Fig 1 and additional evaluator feedback 4 Oct 2021
0.5.3	Addressed open comments from evaluator feedback 11 Oct 2021
0.6.0	Remove iOS information. This ST is now based on Android only
0.6.2	Addresses findings of a review by the evaluator's QA team
0.7.0	Removed File Encryption Module (MOD_FE) and added CAVP certs
0.7.1	Removed remaining MOD_FE related text
1.0.0	Added specific guidance ref in physical scope of the TOE (TOE description)
1.1.0	Updates to FDP_DEC_EXT.1, FPT_TUD_EXT.2 and FTP_DIT_EXT.1.1
1.1.1	Added Enable/disable call history under FMT_SMF.1
1.1.2	Updated FPT_LIB_EXT.1 and FDP_DEC_EXT.1.2
1.2.0	Added PAA in operating environment. Updated FPT_LIB_EXT.1
1.2.1	Updated FPT_LIB_EXT.1 and FMT_SMF.1
1.2.2	FPT_TUD_EXT.2 added heading. Added TSS label FCS_TLSC_EXT.5
1.2.3	Revised FCS_CKM.1/2 (safe-primes)
1.2.4	Synchronised dates between ST and AGD

1. ST INTRODUCTION

This Security Target (ST) specifies the requirements for the Cellcrypt Android Mobile Client Target of Evaluation (TOE) for evaluation and certification under the Common Criteria (CC).

The format and content are in accordance with Common Criteria assurance class ASE and the requirements of the Protection Profiles, Functional Packages, PP-Modules and PP-Configurations stated in Section 2.

1.1. ST and TOE Reference

Table 1 ST and TOE Reference information

Attribute	Description
ST Title	Cellcrypt Android Mobile Client version 4.40 Security Target
ST Version	1.2.3
ST Reference	ST-FED-MCL-And-1
ST Date	Sep 19, 2022
TOE Title	Cellcrypt Android Mobile Client
TOE Version (Android)	4.40
TOE Developer	Cellcrypt
CC Version	Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 5, April 2017

1.2. TOE Overview

Cellcrypt Android Mobile Client is a secure multimedia application for Android smartphones. It implements end-to-end encryption and authentication of voice, video, text messages and file attachments between two or more users of Cellcrypt Android Mobile Client and other compatible applications. The Cellcrypt system comprises a handset software application (Cellcrypt Android Mobile Client, i.e. the TOE) and the back-end support infrastructure (Cellcrypt Server). The TOE is the handset software application, Cellcrypt Android Mobile Client, on a specific hardware platform (described below).

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

Mutually authenticated connection set-up ensures that only mobile phones on which the TOE runs can participate in secure sessions with the Cellcrypt Server, and that the users of the TOE can be assured to always connect to a legitimate Cellcrypt server. End-to-end encryption is achieved through the creation and use of session-unique encryption/decryption keys used by the TOE to encrypt and decrypt voice traffic, messages, and attachments. Long-term static keys and other sensitive user data are stored by the TOE in an encrypted database (SQLCipher) with the SQLCipher master key being protected by the operating system.

The following prerequisites must apply in the use of the TOE:

- The Android mobile platform is the Samsung Galaxy S20 running Android 11.0 on a Qualcomm Snapdragon 865 ARMv8 processor with Processor Algorithm Accelerators (PAA).
- The TOE runs on a NIAP-validated configuration of a mobile platform (including VPN), as defined by the Protection Profile for Mobile Device Fundamentals. The mobile platform is outside the scope of the evaluation.
- ESC Server, as defined by the PP-Module for Enterprise Session Controller (ESC) is outside the scope of this evaluation.
- The TOE operates 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.

The non-TOE components required by the TOE are the following:

- CRL or OCSP server for use in the verification of X.509 certificates.
- Cellcrypt Server for client authentication and other services e.g. SIP, messaging/attachments and check for updated software.

1.3. TOE Description

The Target of Evaluation (TOE) is the Cellcrypt Android Mobile Client application (Figure 1), which runs on Android 11. The Cellcrypt Android Mobile Client application is a software cryptographic application for smartphones. The core function of the TOE is to allow users' voice and video calls to be encrypted with end-to-end security.

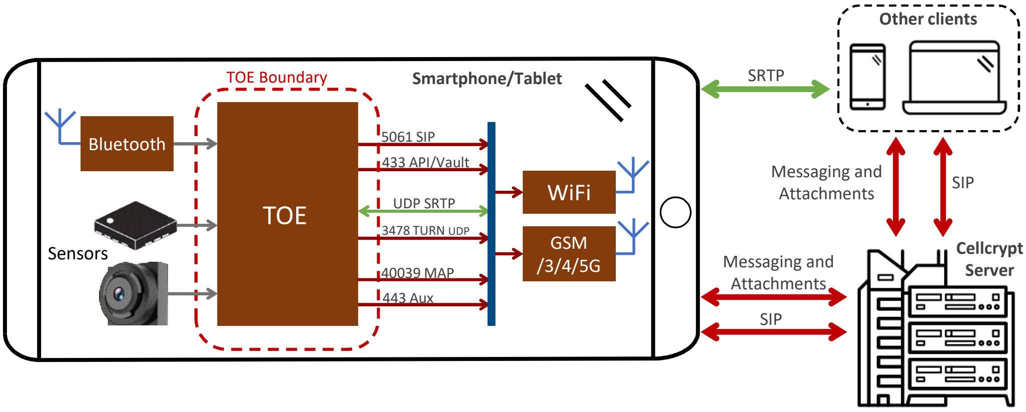

Figure 1 TOE Boundary

The physical scope of the TOE comprises of the following:

- The TOE Software, i.e. the Cellcrypt Android Mobile Client Version (Android) 4.40.

- TOE Security Guidance: Common Criteria Guidance - Cellcrypt Android Mobile Client, AG-FED-MCL-And-1, Version 1.1.2, Sep, 19 2022.

The logical scope of the TOE comprises of the following:

- Authenticated call set-up with the Cellcrypt Server.
- End-to-end encryption of secure voice and video traffic.
- Security management functions restricted to authorized personnel.
- Protection measures for ensuring the integrity and authenticity of the TOE.

The TOE uses X.509 Certificates for mutual authentication on the trusted channel between itself and the Cellcrypt Server. The validity of the X.509 certificates is checked by querying a CRL or an OCSP responder. The TOE uses TLSv1.2 protocol to protect all communications between itself and the Cellcrypt Server from modification and disclosure. In addition to the X.509 Certificate authentication, the TOE also authenticates the user to the Cellcrypt Server using a password. The TOE does not store the authentication password but requests the user to enter it each time it is required.

The TOE achieves end-to-end encryption using an SDES-SRTP trusted channel. The keys for the SDES-SRTP trusted channel are protected by the TLS/SIP channel during key establishment.

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

1.4. TOE Cryptography

All TOE cryptography is performed by the Cellcrypt CCoreV4 FIPS 140-2 validated crypto module. CAVP Certificate references are given in Table 2. The TOE cryptographic support includes functions supporting key management, encryption and decryption, random number generation, digital signatures, secure hashing, and keyed secure hashing. Cryptographic protocol support includes TLS.

Table 2 CAVP Certificate References

Algorithms	Options	Certificates
AES (FIPS 197)	Modes: CTR, CBC, GCM (SP 800-38D) Key lengths: 128, 256 bits	CAVP: A1999
SHA-1 (FIPS 180-4)	Hash lengths: 160 bits	CAVP: A1999
SHA2 (FIPS 180-4)	Hash lengths: 256, 384, 512 bits	CAVP: A1999
HMAC (FIPS 198)	Hash lengths: 160, 256, 384, 512 bits	CAVP: A1999
RSA (FIPS 186-4)	KeyGen, SigGen, SigVer Key length: 2048 bits	CAVP: A1999
DH	KeyExch	CAVP: A1999

Algorithms	Options	Certificates
	Key Length: 2048 bits	
KAS-ECC (SP 800-56Ar1)	KeyExch Curves: P-256, P-384	CAVP: A1999
ECDSA (FIPS 186-4)	KeyGen, SigGen, SigVer Curves P-256, P-384	CAVP: A1999

2. CONFORMANCE CLAIMS

2.1. CC Conformance Claims

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

- Common Criteria for Information Technology Security Evaluation Part 1, Version 3.1, Revision 5, April 2017
- Common Criteria for Information Technology Security Evaluation Part 2, Version 3.1, Revision 5, April 2017: Part 2
- Common Criteria for Information Technology Security Evaluation Part 2, Version 3.1, Revision 5, April 2017: Part 3

The ST is Common Criteria Part 2 extended conformant and Common Criteria part 3 extended conformant.

The ST is package conformant to: **none**. The Security Assurance Components are taken exactly from the Protection Profiles, Protection Profile Modules and Functional packages. They do not constitute an Evaluation Assurance Level or other assurance package.

2.2. Protection Profile Conformance

The TOE claims exact conformance to:

- Protection Profile for Application Software Version 1.3, 2019-03-01 (AppPP)
- PP-Module for Voice and Video over IP (VVoIP) Version 1.0, 2020-10-28 (MOD_VVoIP)
- Functional Package for Transport Layer Security (TLS) Version 1.1, 2019-03-01 (TLS-PKG)

Protection Profile Conformance is claimed in accordance with the following:

- PP-Configuration for Application Software and Voice/Video over IP (VVoIP) Endpoints (CFG_APP-VVoIP_V1.0)

The following NIAP Technical Decisions (TD) apply to the AppPP, MOD_VVoIP and TLS-PKG. Their applicability to the evaluation was determined based on whether the TD is current (i.e. not superseded) and whether the SFRs referenced by the TD are included in the ST:

Table 3: Technical Decisions

TD	Applies	PP/MOD	Exclusion Rationale
0601: X.509 SFR Applicability in App PP	Yes	AppPP	
0600: Conformance claim sections updated to allow for MOD_VPNC_V2.3	No	AppPP	The ST claims no conformance to MOD_VPNC
0598: Expanded AES Modes in FCS_COP for App PP	Yes	AppPP	
0589: Reliable Time for VVoIP Software TOEs	Yes	MOD_VVoIP	
0588: Session Resumption Support in TLS package	No	TLS-PKG	FCS_TLSS_EXT.1.1 is not claimed.
0582: PP-Configuration for Application Software and Virtual Private Network (VPN) Clients now allowed	Yes	AppPP	
0561: Signature verification update	Yes	AppPP	
0554: iOS/iPadOS/Android AppSW Virus Scan	Yes	AppPP	
0548: Integrity for installation tests in AppSW PP 1.3	Yes	AppPP	
0544: Alternative testing methods for FPT_AEX_EXT.1.1	Yes	AppPP	
0543: FMT_MEC_EXT.1 evaluation activity update	Yes	AppPP	
0519: Linux symbolic links and FMT_CFG_EXT.1	No	AppPP	Applies only to Linux.
0515: Use Android APK manifest in test	Yes	AppPP	
0510: Obtaining random bytes for iOS/macOS	No	AppPP	Applies only to iOS.
0498: Application Software PP Security Objectives and Requirements Rationale	Yes	AppPP	
0495: FIA_X509_EXT.1.2 Test Clarification	No	AppPP	This TD only applies when conformance to the PP-Module for VPN Client Version 2.1 is also claimed.
0465: Configuration Storage for .NET Apps	No	AppPP	Applies only to Windows.
0445: User Modifiable File Definition	Yes	AppPP	
0437: Supported Configuration Mechanism	Yes	AppPP	

TD	Applies	PP/MOD	Exclusion Rationale
0435: Alternative to SELinux for FPT_AEX_EXT.1.3	No	AppPP	Applies only to SELinux.
0434: Windows Desktop Applications Test	No	AppPP	Applies only to Windows.
0427: Reliable Time Source	No	AppPP	Only applicable to VPN apps.
0416: Correction to FCS_RBG_EXT.1 Test Activity	No	AppPP	The TOE only invokes the platform provided DRBG functionality.
0513: CA Certificate loading	Yes	TLS-PKG	
0499: Testing with pinned certificates	Yes	TLS-PKG	
0469: Modification of test activity for FCS_TLSS_EXT.1.1 test 4.1	No	TLS-PKG	N/A Only for apps with servers
0442: Updated TLS Ciphersuites for TLS Package	Yes	TLS-PKG	

2.3. Conformance Rationale

This security target claims exact Protection Profile Conformance in accordance with the following:

- PP-Configuration for Application Software and Voice/Video over IP (VVoIP) Endpoints, Version 1.0, 28 October 2020 [CFG_APP-VVoIP_V1.0].

  The PP-Configuration includes the following:

  - Protection Profile for Application Software Version 1.3, 2019-03-01 [AppPP],
  - PP-Module for Voice/Video over IP (VVoIP) Endpoints Version 1.0, 2020-10-28 [MOD_VVoIP],

Additionally, this Security Target claims exact conformance to the following:

- Functional Package for Transport Layer Security (TLS) Version 1.1, 2019-03-01 [TLS-PKG].

The security problem definition, security objectives and security requirements in this Security Target are all taken from the Protection Profile, the PP-Modules, and the Functional Package, applied in accordance with the applicable PP-Configurations. Only operations defined therein are performed on the security functional and security assurance components.

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 [MOD_VVoIP] and is reproduced for the convenience of the reader. [TLS-PKG] does not state additional Security Problem Definition elements.

3.1. Threats Addressed by the TOE

Table 4: Threats Addressed by the TOE

Threat	Description
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_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.MEDIA_DISCLOSURE	An attacker can use the encrypted variable rate vocoder frames to their advantage to decode transmitted data.

3.2. Organizational Security Policies

The [AppPP], [MOD_VVoIP] and [TLS-PKG] do not specify any organizational security policies.

3.3. Assumptions

Table 5: Assumptions on the Operational Environment

Assumption	Description
A.PLATFORM	The TOE relies upon a trustworthy computing platform with a reliable time clock 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 wilfully 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, wilfully negligent or hostile, and administers the software within compliance of the applied enterprise security policy.
A.UPDATE_SOURCE	It is assumed that TOE software/firmware updates will be made available on either the call control server that the TOE connects to or a separate file server managed by the organization.

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], [MOD_VVoIP] and are reproduced for the convenience of the reader. There are no additional security objectives stated in [TLS-PKG].

4.1. Security Objectives for the TOE

Table 6: 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 behaviour 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.

Security Objective	Description
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.
O.ENCRYPTION	To prevent data disclosure from decryption, conformant TOEs will transmit and store sensitive data using mechanisms that provide adequate protections.
O.NO_UNATTENDED_TRANSMISSION	To prevent undetected transmissions, conformant TOEs will not transmit unattended voice or video data when streaming media is not in use.
O.TOE_ADMINISTRATION	To support the enforcement of other security functionality, a conformant TOE will provide a management capability that allows for configuration of the TSF.

4.2. Security Objectives for the Operational Environment

Table 7: 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 wilfully 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, wilfully negligent or hostile, and administers the software within compliance of the applied enterprise security policy.
OE.UPDATE_SOURCE	The operational environment will have TOE software/firmware made available on either the call control server that the TOE connects to or a separate file server managed by the organization.

4.3. Security Objectives Rationale

The security objectives rationale is identical to [AppPP], [MOD_VVoIP] and [TLS-PKG]. It is not repeated herein.

5. SECURITY REQUIREMENTS

This section identifies the Security Functional Requirements for the TOE, including the security requirements rationale (minus the dependency rationale) and the security assurance requirements. 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 5, dated: April 2017, all applicable international interpretations, and [AppPP], [MOD_VVoIP] and [TLS-PKG].

5.1. Conventions

The CC defines operations on Security Functional Requirements: iterations, assignments, selections, assignments within selections and refinements. This document uses the following typographic conventions to identify the operations performed in the ST:

- Iterations following the precise notation used in the source of the relevant SFR, for example FCS_COP.1(1) or FPT_COP.1/SRTP;
- Assignment: Indicated with ***bold italics*** text;
- Selection: Indicated with underlined text;
- Assignment within a Selection: Indicated with ***underlined bold italics*** text.

Formatting conventions for operations performed by the [AppPP], [MOD_VVoIP], or [TLS-PKG] 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 are summarized in *Table 8* and stated in the following subsections.

Table 8: Security Functional Requirements

Class	SFR	PP/MOD	Required Component
FCS: Cryptographic Support	FCS_RBG_EXT.1	AppPP	Random Bit Generation Services
	FCS_CKM.1(1)	AppPP	Cryptographic Asymmetric Key Generation
	FCS_CKM.2	AppPP	Cryptographic Key Establishment
	FCS_CKM_EXT.1	AppPP	Cryptographic Key Generation Services
	FCS_STO_EXT.1	AppPP	Storage of Credentials

Class	SFR	PP/MOD	Required Component
	FCS_COP.1(1)	AppPP	Cryptographic Operation - Encryption/Decryption
	FCS_COP.1(2)	AppPP	Cryptographic Operation - Hashing
	FCS_COP.1(3)	AppPP	Cryptographic Operation - Signing
	FCS_COP.1(4)	AppPP	Cryptographic Operation - Keyed-Hash Message Authentication
	FCS_COP.1/SRTP	MOD_VVoIP	Cryptographic Operation (Encryption/Decryption for SRTP)
	FCS_SRTP_EXT.1	MOD_VVoIP	Secure Real-Time Transport Protocol
	FCS_TLS_EXT.1	TLS-PKG	TLS Protocol
	FCS_TLSC_EXT.1	TLS-PKG	TLS Client Protocol
	FCS_TLSC_EXT.2	TLS-PKG	TLS Client Support for Mutual Authentication
	FCS_TLSC_EXT.3	TLS-PKG	TLS Client Support for Signature Algorithms Extension
	FCS_TLSC_EXT.5	TLS-PKG	TLS Client Support for Supported Groups Extension
FCO: Communications	FCO_VOC_EXT.1	MOD_VVoIP	Fixed-Rate Vocoder
FDP: User Data Protection	FDP_IFC.1	MOD_VVoIP	Subset Information Flow Control
	FDP_IFF.1	MOD_VVoIP	Simple Security Attributes
	FDP_DEC_EXT.1	AppPP	Access to Platform Resources
	FDP_NET_EXT.1	AppPP	Network Communications
	FDP_DAR_EXT.1	AppPP	Encryption Of Sensitive Application Data
FIA: Identification and Authentication	FIA_X509_EXT.1	AppPP	X.509 Certificate Validation
	FIA_X509_EXT.2	AppPP	X.509 Certificate Authentication
FMT: Security Management	FMT_MEC_EXT.1	AppPP	Supported Configuration Mechanism
	FMT_SMF.1	AppPP	Specification of Management Functions
	FMT_SMF.1/VVoIP	MOD_VVoIP	Specification of Management Functions (VVoIP Communications)
	FMT_CFG_EXT.1	AppPP	Secure by Default Configuration
FPR: Privacy	FPR_ANO_EXT.1	AppPP	User Consent for Transmission of Personally Identifiable Information

Class	SFR	PP/MOD	Required Component
FPT: Protection of the TSF	FPT_API_EXT.1	AppPP	Use of Supported Services and APIs
	FPT_AEX_EXT.1	AppPP	Anti-Exploitation Capabilities
	FPT_IDV_EXT.1	AppPP	Software Identification and Versions.
	FPT_TUD_EXT.1	AppPP	Integrity for Installation and Update
	FPT_TUD_EXT.2	AppPP	Integrity for Installation and Update
	FPT_LIB_EXT.1	AppPP	Use of Third Party Libraries
FTA: TOE Access	FTA_SSL.3/Media	MOD_VVoIP	TSF-Initiated Termination (Media Channel)
FTP: Trusted Path/Channel (FTP)	FTP_DIT_EXT.1	MOD_VVoIP	Protection of Data in Transit
	FTP_ITC.1/Control	MOD_VVoIP	Inter-TSF Trusted Channel (Signaling Channel)
	FTP_ITC.1/Media	MOD_VVoIP	Inter-TSF Trusted Channel (Media Channel)

5.2.1. Cryptographic Support (FCS)

5.2.1.1. 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.

5.2.1.2. 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** [

- ***[RSA schemes]*** *using cryptographic key sizes of* ***[2048-bit or greater]*** *that meet the following* ***FIPS PUB 186-4, "Digital Signature Standard (DSS), Appendix B.3***",
- ***[ECC schemes]*** *using* ***["NIST curves" P-256, P-384 and [no other curves]]*** *that meet the following:* ***[FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4]***,
- ***[FFC schemes****]* *using* ***"safe-prime" groups*** *that meet the following:* ***NIST Special Publication 800-56A Revision 3, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" and [RFC 7919] ,***

].

FCS_CKM.1.2(3) The TSF shall generate salts using a RBG that meets FCS_RGB_EXT.1 and with entropy corresponding to the security strength selected for PBKDF in FCS_CKM.1.1(3).

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: [

- ***[RSA-based key establishment schemes]*** *that meet the following:* ***RSAES-PKCS1-v1_5 as specified in Section 7.2 of RFC 8017, "Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1" ,***
- ***[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"]***,
- *[**FFC schemes**] **using "safe-prime" groups*** *that meet the following:* ***NIST Special Publication 800-56A Revision 3, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" and [RFC 7919],***

].

5.2.1.4. FCS_CKM_EXT.1 Cryptographic Key Generation Services

FCS_CKM_EXT.1.1 The application shall [

- *implement asymmetric key generation*

].

5.2.1.5. 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 [**database encryption key**],*
- *implement functionality to securely store [**X.509 certificates and associated private keys**] according to [FCS_COP.1(1)]*

] to non-volatile memory.

5.2.1.6. 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,*
- *AES-GCM (as defined in NIST SP 800-38D) mode,*
- *AES-CTR (as defined in NIST SP 800-38A) mode,*

] and cryptographic key sizes [*128-bit, 256-bit*][1].

5.2.1.7. FCS_COP.1(2) Cryptographic Operation – Hashing

FCS_COP.1.1(2) The **application** shall perform *cryptographic hashing* services in accordance with a specified cryptographic algorithm [

- *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.

5.2.1.8. 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 [no other curves] that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 5*

].

5.2.1.9. 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,*
- *SHA-512*

] with key sizes [***256-bits***] and message digest sizes 256 and [*160, 384, 512*] bits that meet the following: FIPS Pub 198-1 *The Keyed-Hash Message Authentication Code* and FIPS Pub 180-4 *Secure Hash Standard*.

[1] In accordance with TD0598

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

FCS_COP.1.1/SRTP: 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), AES-GCM (as defined in NIST SP 800-38D)*] and cryptographic key sizes [*128-bit, 256-bit*].

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_TLS_EXT.1 TLS Protocol

The product shall implement [

- *TLS as a client,*

].

5.2.1.12. FCS_TLSC_EXT.1 TLS Client Protocol

FCS_TLSC_EXT.1.1 The product shall implement TLS 1.2 (RFC 5246) and [*no earlier TLS versions*] as a client that supports the cipher suites [

- *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*

] and also supports functionality for [

- *Mutual authentication*

][2].

FCS_TLSC_EXT.1.2 The product shall verify that the presented identifier matches the reference identifier according to RFC 6125.

FCS_TLSC_EXT.1.3 The product shall not establish a trusted channel if the server certificate is invalid [

- *With no exceptions*

].

5.2.1.13. FCS_TLSC_EXT.2 TLS Client Support for Mutual Authentication

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

5.2.1.14. FCS_TLSC_EXT.3 TLS Client Protocol for Signature Algorithm Extension

FCS_TLSC_EXT.3.1 The product 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.5 TLS Client Support for Supported Groups Extension

FCS_TLSC_EXT.5.1 The product shall present the Supported Groups Extension in the Client Hello with the supported groups [

- *secp256r1,*
- *secp384r1*

].

5.2.2. Communications (FCO)

5.2.2.1. FCO_VOC_EXT.1 Fixed-Rate Vocoder

FCO_VOC_EXT.1.1 The TSF shall transmit voice media using a constant bit rate vocoder.

5.2.3. User Data Protection (FDP)

5.2.3.1. FDP_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*].

[2] In accordance with TD0442

5.2.3.2. FDP_IFF.1 Simple Security Attributes

FDP_IFF.1.1 The TSF shall enforce the [*media transmission policy*] based on the following types of subject and information security attributes: [*TOE hook state, VVoIP call connection status, and VVoIP call control server status*].

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 a VVoIP call control server],***
- *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 [*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*].

5.2.3.3. FDP_DEC_EXT.1 Access to Platform Resources

FDP_DEC_EXT.1.1 The application shall restrict its access to [

- *network connectivity,*
- *camera,*
- *microphone,*
- *Bluetooth*

].

FDP_DEC_EXT.1.2 The application shall restrict its access to [

- ***[Android file system]***

].

5.2.3.4. FDP_NET_EXT.1 Network Communications

FDP_NET_EXT.1.1 The application shall restrict network communication to [

- *user-initiated communication for [****connecting to a SIP server, connecting to a VVoIP endpoint, checking for updates****],*
- *[certificate validation using CRL, certificate validation with OCSP, 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 [

- *protect sensitive data in accordance with FCS_STO_EXT.1*

] 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, and that any path constraints are met
- The application shall validate that any CA certificate includes caSigning purpose in the key usage field
- The application shall validate the revocation status of the certificate using *[OCSP as specified in RFC 6960, CRL as specified in RFC 5280 Section 6.3, an OCSP TLS Status Request Extension (i.e. OCSP stapling) as specified in RFC 6066*]
- 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 EKU 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 EKU 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 EKU 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 EKU 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 EKU 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*][3].

5.2.5.2. FMT_SMF.1 Specification of Management Functions

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

- [***Register and unregister with a SIP server,***
- ***Load X.509 Certificates and private keys,***
- ***Select X.509 Certificate,***
- ***Check for software updates,***
- ***Specify ESC server URL,***
- ***Select TLS cipher suites,***
- ***Enable/disable call history***]

].

5.2.5.3. FMT_SMF.1/VVoIP Specification of Management Functions (VVoIP Communications)

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

- Ability to *[register the TOE to an ESC [manually]];*
  [
  - *Ability to configure the termination period for idle calls;*
  - *Ability to specify the vocoder used;*
  - *Ability to specify the port to be used for SRTP communications]*

].

5.2.5.4. 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 binaries and data files from modification by normal unprivileged users.

[3] In accordance with TD0437

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 use only 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 built with stack-based buffer overflow protection enabled.

5.2.7.3. FPT_IDV_EXT.1 Software Identification and Versions

FPT_IDV_EXT.1.1 The application shall be versioned with [***[version information encoded in the installation file name and a visible version string accessible in the Settings menu]***].

5.2.7.4. 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 [*provide the ability*] to query the current version of the application software.

FPT_TUD_EXT.1.3 The application shall not download, modify, replace or update its own binary code.

FPT_TUD_EXT.1.4 ~~The a~~Application ~~installation package and its~~ updates shall be digitally signed such that ~~it's~~ the application platform can cryptographically verify them prior to installation[4].

FPT_TUD_EXT.1.5 The application is distributed [*as an additional software package to the platform OS*].

FPT_TUD_EXT.2.1 The application shall be distributed using the format of the platform-supported package manager.

5.2.7.5. FPT_TUD_EXT.2 Integrity for Installation and Update

FPT_TUD_EXT.2.2 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.2.3 The application installation package shall be digitally signed such that its platform can cryptographically verify them prior to installation.

5.2.7.6. FPT_LIB_EXT.1 Use of Third Party Libraries

FPT_LIB_EXT.1.1 The application shall be packaged with only **[**

- ***libc++_shared.so***
- ***libccopenssl-fips.so***
- ***libsc3_opus.so***
- ***libsqlcipher.so***
- ***libvphone.so***

].

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 [*inactivity longer than* [*__[30]__ seconds, an administrator configurable interval*]].

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 [data] with **TLS as defined in the TLS Package** and [Secure Real-Time Transport Protocol (SRTP)]*

] between itself and another trusted IT product[5].

[4] In accordance with TD0561

[5] In accordance with TD0601

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 [*Session Initiation Protocol (SIP)*]** to provide a trusted communication channel between itself and **a VVoIP call control server** 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 VVoIP call control server*]** 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

The Security Assurance Requirements applicable to the TOE are those stated in Sect. 5.2 of [AppPP] and in Sect. 2.2 of [CFG_APP-VVoIP_V1.0]. Neither [MOD_VVoIP] nor [TLS-PKG] defines additional Security Assurance Requirements. For the sake of compactness, the Security Assurance Requirements are not repeated herein.

5.4. Security Requirements Rationale

The security requirements rationale is identical to [AppPP], [MOD_VVoIP] and [TLS-PKG]. It is not repeated herein.

6. TOE SUMMARY SPECIFICATION

This chapter identifies and describes how the Security Functional Requirements are met by the TOE. CAVP Certificate references for all cryptographic algorithms are given in Table 2.

Table 9: TOE Summary Specification Description

Requirement	Rationale
FCS_RBG_EXT.1	The TOE invokes the platform provided DRBG for all random bit generation. Android uses the Java SecureRandom class to access the system DRBG which is FIPS 140-2 validated. The random bit generation is used by the following functions of the TOE: • Generation of asymmetric and symmetric cryptographic keys, • Generation of Initialization Vectors for symmetric cryptography.
FCS_CKM.1(1) FCS_CKM.2 FCS_CKM_EXT.1	The TOE generates asymmetric RSA keys and EC keys in accordance with FIPS PUB 186-4 for TLS key establishment. RSA key length is 2048 bits and EC keys lengths are 256 and 384 bits with NIST curves P-256 and P-384, respectively. The EC and FFC key establishment scheme meets SP800-56A. The keys are used for the following: • Digital signature computation, • TLS Protocol implementation, and • X.509 certificate-based authentication.
FCS_COP.1(1)	The TOE implements AES for encryption and decryption on TLS and SRTP links. For TLS the TOE implements CBC and GCM modes. For SRTP, the TOE implements CTR, and GCM modes, all using 128-bit and 256-bit keys.
FCS_COP.1(2)	The TOE implements the following hashing algorithms: SHA-1, SHA2-256, SHA2-384, SHA2-512.
FCS_COP.1(3)	The TOE implements RSA and ECDSA signature generation and verification in the CCoreV4 module. RSA keys of 2048-bits are supported. ECDSA keys using P-256 and P-384 are supported.
FCS_COP.1(4)	The TOE implements HMAC message authentication in the Cellcrypt CCoreV4 FIPS 140-2 library. HMAC-SHA-1, HMAC-SHA-256, and HMAC-SHA-384 are supported.

Requirement	Rationale
FCS_SRTP_EXT.1 FCS_COP.1/SRTP 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 CCoreV4 module 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 user credentials and other sensitive information is stored in an encrypted database (SQLCipher). The following data is stored in the database: • User profile • User keys and certificates • User contacts The TOE invokes the Android Keystore API to store the AES 256-bit database encryption key. The database is encrypted according to FCS_COP.1(1). The MODE_PRIVATE flag is also set on the database file.

Requirement	Rationale
FCS_TLS_EXT.1 FCS_TLSC_EXT.1 FCS_TLSC_EXT.2 FCS_TLSC_EXT.3 FCS_TLSC_EXT.5	The TOE implements TLS v1.2 as specified in RFC 5246 and supports the following ciphersuites: • TLS_RSA_WITH_AES_128_CBC_SHA256, TLS_RSA_WITH_AES_256_CBC_SHA256, TLS_DHE_RSA_WITH_AES_128_CBC_ SHA256 and TLS_DHE_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246 • 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, TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256, and TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 The TOE establishes the reference identifier by parsing the DNS Name 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 wildcards in the DNS name of the server certificate. When TLS mutual authentication is used, the client sends to the TOE a client-certificate as part of the TLS handshake. The TOE validates the certificate to establish the authenticity of the client. Certificate pinning is supported. Server certificate is matched based on its public key. The pinned server public key can be updated after a successful sign-in with user password. The TOE presents the supported Elliptic Curves Extension in the Client Hello message with the P-256 and P-384 curves. This is the default TOE behaviour 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 behaviour and cannot be modified.
FCO_VOC_EXT.1	The TOE uses the Opus, G.711(PCMA) and G.711(PCMU) vocoders to transmit voice media. The Opus vocoder generates a constant bit-rate stream of 48 Kbps and the G.711 codecs generate a constant bit-rate stream of 8 Kbps. In low-bandwidth mode the TOE uses Opus at 12 Kbps.

Requirement	Rationale
FDP_DEC_EXT.1	The TOE accesses network connectivity, camera, microphone and Bluetooth resources. Bluetooth is an optional setting for hands-free audio. The TOE also accesses its database through the Android file system and can read from the Android filesystem for importing the root certificate and user's .p12 credentials file. Here is the full list of permissions requested by the app: ACCESS_NETWORK_STATE ACCESS_WIFI_STATE BLUETOOTH (android:maxSdkVersion="30") BLUETOOTH_CONNECT (usesPermissionFlags="neverForLocation" tools:targetApi="s") BROADCAST_STICKY INTERNET MODIFY_AUDIO_SETTINGS RECEIVE_BOOT_COMPLETED VIBRATE WAKE_LOCK FOREGROUND_SERVICE USE_BIOMETRIC USE_FULL_SCREEN_INTENT READ_PHONE_STATE CAMERA RECORD_AUDIO READ_EXTERNAL_STORAGE WRITE_EXTERNAL_STORAGE RECEIVE REQUEST_IGNORE_BATTERY_OPTIMIZATIONS C2D_MESSAGE (for Android push notifications)
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 a SIP server • Communicating with a VVoIP endpoint. • Communication with an authentication server (API) • Communication with a message attachment server (Vault) • Communicating with an Update check server. The TOE automatically initiates the following network communications: • CRL certificate validation. • OCSP certificate validation. • Inactive call timeout setting update from the configuration server.
FDP_DAR_EXT.1	Sensitive data consisting of user credentials and contact information are stored in an encrypted database according to FCS_COP.1(1) with keys stored according to FCS_STO_EXT.1.
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 client-side certificate used for authenticating the client is sent by the client during the TLS handshake. The TOE performs validity checks on the CA path and confirms that either the SubjectAltName or CN match what was provided on the distant connection certificate. The TSF also performs CRL and OCSP validity checks on the server 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 TOE 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.

Requirement	Rationale
FMT_MEC_EXT.1	The TOE stores its security-related configuration settings in an encrypted SQCipher database stored at /data/data/com.cellcrypt.federal/databases/[identifier] where identifier is based on the user ID. The encrypted database key is generated and stored using the Android Key Store provider (KeyStore.getInstance("AndroidKeyStore")). Other general settings are stored in the Android shared-prefs.
FMT_SMF.1	The TOE has a GUI which allows users to perform the following management functions: • Enable/disable transmission of call statistics, logs using slider switches. • Specify the ESC server URL when signing in. • Select and Load X.509 Certificates and private keys from the file system. • Check for software updates. • Select the TLS cipher-suites from a list (check boxes). • Enable/disable call history (slider switch) These settings are stored in the encrypted SQCipher database as described in FMT_MEC_EXT.1.
FMT_SMF.1/VVoIP	The TOE provides the following VVoIP management settings: • Manually register and unregister to the ESC (slider control switch) • Configure the termination period for idle calls (text edit box control for seconds) • Specify the vocoder from a list of supported vocoders (radio buttons) • Select the SRTP cipher-suites from a list (check boxes) • Specify the SRTP port (text edit box control)
FMT_CFG_EXT.1	The TOE does not contain any default credentials when it is installed. TOE credentials (certificates and keys) must be configured before the TOE can connect to the ESC. User credentials (username and password) must be registered with the ESC before the user can make secure calls.
FPR_ANO_EXT.1	The TOE does not transmit PII.

Requirement	Rationale
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_LIB_EXT.1	The TOE is packaged with following libraries: • libc++_shared.so - standard C++ libraries. • libccopenssl-fips.so – Cellcrypt library consisting of OpenSSL with Cellcrypt’s CCoreV4 FIPS 140-2 validated cryptographic module https://csrc.nist.gov/projects/cryptographic-module-validation-program/certificate/4178. • libsc3_opus.so – Cellcrypt library consisting of opus codec. • libsqlcipher.so – SQLCipher database library. • libvphone.so – Cellcrypt library consisting of openh264, mp3lame, pjsip, libq, srtp and rxcpp libraries.
FPT_AEX_EXT.1	The TOE is built with the -fstack-protector-all compiler flag which enables stack-based buffer overflow protection. ASLR is enabled using the -fPIC compiler flag.
FPT_IDV_EXT.1	TOE version information is encoded in the installation file name and is visible through the version string accessible in the Settings menu of the underlying platform. The TOE version is a two-part string of a TOE version and release concatenated and separated by full stops, i.e. as in 3.182.

Requirement	Rationale
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. The TOE allows the user to query the application's version in the user settings. Updates to the TOE are distributed in the Android Package (APK) format (.apk). 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.
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 Cellcrypt CCoreV4 FIPS 140-2 validated crypto module to perform cryptographic operations. The TOE uses TLSv1.2 (FCS_TLSC_EXT) to protect the SIP communications.

Requirement	Rationale
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. There are no partial updates, any update shall include the entire TOE. Each customer identified security officer is notified via email when a security update is available. The updates are distributed in Android Package (APK) format (.apk). TOE APK files are digitally signed with the Cellcrypt private key. The signature is verified by the Android installation routine. The time between disclosure of a vulnerability and availability of a security update varies from two weeks to 90 days. Cellcrypt gets weekly OS updates/vulnerability notifications via OS subscriptions and our customers are notified of the same via their own subscriptions. Support contracts can also be arranged allowing Cellcrypt to notify customers directly via email. Third-party libraries are treated in the same way and unlicenced open-source third-party library vulnerabilities are monitored via MITRE CVE and NIST NCD feeds.

APPENDIX A – TERMINOLOGY AND ACRONYMS

Table 10: Terminology and Acronyms

Term	Description
AES	Advanced Encryption Standard
ACVP	Automated Cryptographic Validation Protocol
API	Application Programming Interface
APK	Android Package
CA	Certificate Authority
CAVP	Cryptographic Algorithm Validation Program
CBC	Cipher Block Chaining
CC	Common Criteria for Information Technology Security Evaluation
CN	Common Name
CRL	Certificate Revocation List
CTR	Counter mode
DRBG	Deterministic Random Bit Generator
ECDSA	Elliptic Curve Digital Signature Algorithm
FIPS	Federal Information Processing Standard
GUI	Graphical User Interface
GCM	Galois Cipher Mode
HMAC	Hash-based Message Authentication Code
IP	Internet Protocol
IV	Initial Vector
KEK	Key Encryption Key
NIAP	National Information Assurance Partnership
NIST	National Institute of Standards and Technology
OID	Object Identifier
OS	Operating System
PAA	Processor Algorithm Accelerators
PII	Personal Identification Information
PP	NIAP Protection Profiles
PRNG	Pseudo Random Number Generator
RNG	Random Number Generator
RSA	Rivest Shamir Adleman
RTP	Real Time Protocol (RFC 3550)

SAN	Subject Alternative Name
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)
TOE	Target Of Evaluation
TSF	TOE Security Functions
TSS	TOE SFR Summary
VPN	Virtual Private Network
VoIP	Voice over Internet Protocol
VVoIP	Video and Voice over Internet Protocol